olivetti

LABORATORIO RICERCHE ELETTRONICHE

L'UNITÀ CENTRALE ELEA 9003

DESCRIZIONE LOGICO FUNZIONALE

SERVIZIO TECNICO ELETTRONICO ESTERNO

Ottobre 1962

1. PREMESSA SULLA FUNZIONE DI ALCUNI FF

1.1. FF Ni

Questo FF staticizza l'indicazione che una istruzione di 1° programma non può essere eseguita.

Esso viene per esempio disposto in $\alpha.P_5.\bar{\alpha}_2.\bar{\alpha}_p.M_4$ (cioè durante la fase preparatoria di una istruzione di 1° programma):

a) se sono presenti i segnali βy e Δβy (che è un decodificato presente in tutte le istruzioni che richiedono l'uso o la prenotazione del canale esterno di memoria); cioè se il canale esterno risulta già occupato (βy) ed è necessario per l'esecuzione della istruzione attualmente letta (Δβy).

b) Nelle istruzioni di nastro (ΔN) se il GUN risulta già occupato ($\overline{Nf}$); ecc.

L'informazione staticizzata in tale FF è utilizzata per evitare che dopo la lettura di una istruzione del 1° programma che non si poté eseguire, quando si legge di nuovo una istruzione di 1° programma, si passi alla istruzione successiva del programma stesso; si deve invece rileggere l'istruzione non ancora eseguita agendo opportunamente sull'indirizzatore W.

Assolta tale funzione (in $\alpha.P_0$) l'informazione staticizzata nel FF Ni non è più necessaria e perciò lo si ripone in $\alpha.\bar{\alpha}_2.\bar{\alpha}_p.P_3.M_9$.

1.2. FF Ne

In modo perfettamente analogo al FF Ni staticizza l'informazione che una istruzione di 2° programma non può essere eseguita.

Esso viene per esempio disposto in $\alpha_2.P_5.M_4$ (cioè durante la fase preparatoria di una istruzione di 2° programma):

a) Se sono presenti i segnali βy e Δβy.

b) Nelle istruzioni di nastro se il GUN risulta già occupato; ecc.
   E' evidente che se si dispone Ne è impossibile continuare il 2° programma.

   L'informazione staticizzata in tale FF è utilizzata per ricordarsi che si deve rileggere l'istruzione non potuta eseguire quando si riprenderà il 2° programma.

   Assolta tale funzione (in $\alpha.P_0$) il FF Ne è riposto in $\alpha.\alpha_2.P_3.M_9$.

1.3. **FF N_3**

Perfettamente analogo ai precedenti FF Ni ed Ne per quanto riguarda un'istruzione letta sul 3° programma.

Si avrà perciò nel set la condizione $\alpha p.P_5.M_4$; nel reset $\alpha.\alpha p.P_3.M_9$.

1.4. **FF Sb**

Ha la funzione di far riprendere il 2° programma che era stato interrotto per la presenza di Ne; cioé tale FF staticizza l'informazione che è cessata una condizione che poteva avere causato il blocco (cioè il set di Ne) del 2° programma; Sb significa appunto sblocco.

1.5. **FF Sb_3**

Ha una funzione perfettamente analoga ad Sb per quanto riguarda il 3° programma. Viene disposto colle stesse condizioni di Sb.

Viene riposto invece con $\alpha p.P_3.M_9$.

PARTE PRIMA

ISTRUZIONI DI UNITÀ CENTRALE

2. ISTRUZIONE MA — DECODIFICATO ΔAUM

2.1. Considerazioni generali sull'istruzione MA

La MA è una istruzione di trasferimento che impegna gli organi interni dell'Unità Centrale.

Trasferisce da Memoria in Accumulatore l'operando individuato dalla stessa istruzione, lasciandolo scritto in Memoria.

Essa si presenta come successione di 8 caratteri alfanumerici: L_dL_uIIIITF che vengono letti da destra a sinistra.

F è il « carattere di funzione »

T è il carattere indicante l'indirizzo del registro T che dovrà modificare col suo contenuto i 5 caratteri L_uIIII

IIII è un indirizzo di Memoria che, modificato dal contenuto del registro T, individua il reale indirizzo dell'operando in Memoria

L_dL_u sono due caratteri che esprimono la lunghezza dell'operando che verrà estratto da Memoria nella fase esecutiva dell'istruzione.

La MA consta di una fase α di preparazione e di una fase β di esecuzione. Ogni carattere viene operato in un tempo di 10 μ s. detto p.d.c.

Nella fase α si leggono gli otto caratteri precedentemente menzionati e si predispongono gli organi interni per l'esecuzione.

2.2 La fase α

Se l'istruzione MA fa parte di un certo programma, essa verrà individuata nel tempo alla fine dell'istruzione precedente, qualora questa impegni i soli organi interni della macchina.

Più in generale si può dire che la fase α viene avviata dal disporsi del FF φ indicante che gli organi interni della macchina sono disimpegnati e quindi in grado di ricevere altre informazioni. Il FF α, che disponendosi dà inizio alla fase α, può essere attivato oltre che dal segnale φ, dal segnale di « via » tramite il FF Pa all'inizio delle operazioni, o dai segnali 2_p, 3_p che permettono la ripresa dello svolgimento del secondo programma o del programma « on line » quando il primo programma viene interrotto dal segnale di STOP; tali casi particolari verranno analizzati in seguito in un capitolo apposito.

Sequenza delle operazioni che seguono il segnale φ

Il segnale φ ripone il FF KZ e dispone il FF α. Il segnale $\overline{KZ}$ azzera il contatore di periodi Cp e dispone il FF P_0.
Queste operazioni hanno luogo nell'ultimo p.d.c. della fase esecutiva dell'istruzione precedente che impegna gli organi interni dell'Unità Centrale. Con l'attivarsi del FF α e del FF P_0 e l'azzerarsi del contatore Cp ha inizio la fase α dell'istruzione MA.

Durante la fase α dell'istruzione precedente si è scritto in un registro ausiliario J l'indirizzo del carattere di P_2 dell'istruzione stessa; e precisamente:

a) nel registro I se l'istruzione è di 1° programma

b) nel registro M se l'istruzione è di 2° programma

c) nel registro H se l'istruzione è di 3° programma.

Si vedrà che ciò accade nella fase α che stiamo esaminando, e in tutte le fasi α.

Per poter leggere l'istruzione successiva si deve aumentare tale indirizzo di 9; ciò è ottenuto con una conta + 10 ed una conta — 1.

Se si deve invece rileggere l'istruzione stessa (per effetto di Ni, Ne, N_3 si deve fare soltanto una conta + 1.

Si presume che l'istruzione precedente alla MA non sia un salto verificato.

p.d.c. P_0

M_0 il FF α è disposto
M_0 si ripone il FF φ
M_0 si cancella il contenuto dell'indirizzatore W
M_2 si dispone il FF DW (se è presente $\overline{Ni}.\bar{\alpha}_2.\bar{\alpha}_p.\overline{ICV}$)
M_2 si dispone il FF IW con $\overline{KZ}$
M_3 si dispone il FF α r
M_3 si dispone il FF PoR
M_3 si ripone il FF Db
M_4 si trasferisce in W il contenuto del registro I e si predispone la maschera del FF Zc (es.: $\overline{Ce}$ x IWD)
E' selezionato il registro I con $\alpha.\bar{\alpha}_p.\bar{\alpha}_2.\overline{ICV}.P_0$.
E' presente CJW = $\alpha.P_0$.

Se la MA è letta sul 2° programma si seleziona il registro M con $\alpha.\alpha_2.P_0.\overline{ICV}$
Se è letta sul 3° programma si seleziona il registro H con $\alpha.\alpha_p.P_0.ICV$

M_5 si dispone il FF KZ
M_7 l'indirizzatore W conta + 10 se c'è DW
M_7 si ripone IW se c'è Ni
M_8 si ripone il FF Dw
M_9 il contatore Cp conta +1. Il decodificatore di p— pulsi fornisce il segnale P_1
M_9 si ripone il FF α_{37}
M_9 si ripone il FF P_0
M_9 si ripone il FF Sm

p.d.c. P_1

M_0 l'indirizzatore W conta —1 (IWD in IWP) (o + 1 se in $P_0.M_7$ si è riposto IW)
M_0 si dispone il FF Zc (il segnale IWP non ha effetto perché arriva in maschera)
M_0 è presente il segnale pMp di consenso al pilotaggio di $Mp = \alpha_r.$ (AUTO) $.\overline{P_0}.\overrightarrow{A_r}.Zc$
M_2 si cancella il contenuto dei FF F.
M_2 si dispone IW (serve se è stato riposto in $\alpha.P_0.M_7$)
M_3 si dispone il FF Zr.
M_3 si ripongono i FF Up di uscita da Mp
M_3 si ripone il FF PoR
M_3 si ripongono i FF Rm
M_3 nasce il segnale $\overline{IDM}$ con $\overline{Db}.\overline{dn}.CMU$; $CMU \equiv \alpha_r.\overline{PoR}$
M_4 si legge in Mp. Il carattere letto viene rigenerato al successivo mastro M_7 essendo presente: $CMpR \equiv Zc.XMG$; $XMG \equiv \alpha r$
M_4 si staticizza nei FF Up il carattere letto in Mp.
M_4 si trascrive nei FF F il contenuto dei FF Up (cioè il carattere letto in memoria)
M_5 i FF Rm si predispongono sul valore 1
M_7 si azzera il registro di lunghezza
M_9 il contatore Cp conta + 1. Si ottiene il segnale P_2.

p.d.c. P_2

M_0 l'indirizzatore W conta —1 (IWP in IWD)

M_0 si ripone Zc con $\overline{Ce}$.IWP (IWD non ha effetto perchè arriva in maschera)

M_0 scompare il segnale pMp e compare il segnale pMd di consenso al pilotaggio di Md; cioè scompare Zc e compare $\overline{Zc}$ donde pMd $= \alpha_r$. (AUTO) $.\overline{P_0}.\overline{A_r}.\overline{Zc}$

M_0 si azzera l'indirizzatore V e si ripone il FF fV con $\alpha.M_0.P_2.\overline{A_r}$

M_2 si ripone il FF Cn con RAs $= \alpha.P_2.\Delta AUM$

M_3 si ripone il FF Zr

M_3 si ripongono i FF Ud di uscita da Md

M_4 si legge in Md. Il carattere letto viene rigenerato al successivo mastro M_7 CMdR $=$ XMG.$\overline{Zc}$

M_4 si staticizza nei FF Ud il carattere letto in Md

M_7 il contenuto di W passa nel registro I
CWJ (consenso da W a registro J) è presente in $\alpha.P_2$.

E' selezionato il registro I con $\alpha.\overline{\alpha_2}.\overline{\alpha_p}.P_2$.

Se la MA fosse sul 2° programma si avrebbe la selezione del registro M con $\alpha.\alpha_2.P_2$.

Se fosse sul 3° programma si avrebbe la selezione del registro H con $\alpha.\alpha_p.P_2$.

M_8 si trasferisce in V il contenuto dei FF Ud passando attraverso l'UAD.

Il carattere trasferito rimane inalterato nel passaggio attraverso Co per la presenza del segnale $\overline{IDM}$ e nel passaggio attraverso l'UAD perché $R_m = 1$ e da memoria T esce zero mancando il relativo pilotaggio.

I bits a, b, c. d del carattere letto vanno a staticizzarsi in Vd; una rete logica dei bits e ad f scrive in Vu 0 o 5 mentre il FF fv disponendosi o rimanendo riposto, individua il pilotaggio PT_I o PT_{II} dei registri T per il p.d.c. seguente.

In paricolare la scrittura del 5 in Vu (set del d Vu) è ottenuta nei casi: $e.\overline{f}+\overline{e}.f$; il FF fv è disposto se è presente f in uscita dell'UAD.

M_9 si dispone il FFα_{37}. E' presente il segnale PT_I o PT_{II} di consenso al pilotaggio della memoria T_I o T_{II}.

Tali segnali si hanno dato che è presente XAF $= \alpha_{37}.\overline{\varnothing a}$.

Nel caso però che sia $\overline{fv}$ e non si abbia $\overline{V\#d}$ non si ha pilotaggio di alcun registro. Il segnale V # d è presente se nel Vd vi è un carattere proibito; quindi qualsiasi carattere proibito senza bit f indica nell'istruzione registro modificatore inesistente

M_9 si ripone il FF Sa con $\alpha.P_2.\Delta A.\Delta U.\Delta M$

M_9 si ripone il FF As con RAs (vedi a M_2)

p.d.c. P_3

M_0 l'indirizzatore W conta —1 (IWD in IWP)

M_0 si dispone il FF Zc

M_0 scompare il segnale pMd e compare il segnale pMp

M_3 si dispone il FF Zr

M_4 si legge in Mp. Il carattere letto viene rigenerato al successivo mastro M_7

M_4 si staticizza nei FF Up il carattere letto in Mp.

M_4 si legge nel registro T. Il carattere letto viene rigenerato al successivo mastro M_7. Comando di rigenerazione CAR $= \alpha r$

M_4 si staticizza nei FF Au il carattere letto in T

M_4 il carattere letto in Mp attraversa inalterato il complementatore Co per la presenza del segnale $\overline{IDM}$; il carattere uscente da registro T passa pure inalterato attraverso Ca per la presenza del segnale $\overline{IDA}$. Essendo il contenuto dei FF Rm = 1, l'UAD esegue la somma dei due caratteri estratti. Il risultato rimane disponibile nella U.A.D.

M_7 il risultato viene scritto nel registro 0, decade delle unità; è selezionato 0 con: $\alpha_{37}.\overline{P_7}.\overline{P_8}.\overline{\Delta AUP}$ (cioè esclusa la DA) è azzerato 0 essendoci CJZ in αP_3 (cioè 0 è azzerato interrogandolo, l'uscita non va in nessun posto); in αP_4-P_5-P_6 CJZ non c'è più e quindi non si ha più interrogazione di 0.

p.d.c. P_4

M_0 l'indirizzatore V conta + 1 con $\alpha_{37}.\overline{P_3}$
M_0 l'indirizzatore W conta —1 (IWP in IWD)
M_0 si ripone il FF Zc
M_0 scompare il segnale pMp e compare il segnale pMd
M_3 si ripone il FF Zr
M_3 si ripongono i FF Ud di uscita da Md
M_4 si legge in Md. Il carattere letto viene rigenerato al successivo mastro M_7
M_4 si staticizza nei FF Ud il carattere letto in Md
M_4 si legge nel registro T. Il carattere letto viene rigenerato al successivo mastro M_7
M_4 si staticizza nei FF Au il carattere letto in T
M_4 si esegue in UAD la somma dei due caratteri letti. Essi passano inalterati attraverso i rispettivi complementatori Co, Ca per la presenza dei segnali $\overline{IDM}$,$\overline{IDA}$
M_7 si scrive il risultato nel registro 0, decade delle decine.

p.d.c. P_5

M_0 l'indirizzatore V conta +1
M_0 l'indirizzatore W conta —1 (IWD in IWP)
M_0 si dispone il FF Zc
M_0 scompare il segnale pMd e compare il segnale pMp
M_3 si dispone il FF Zr
M_3 si ripongono i FF Up di uscita da Mp
M_4 si legge in Mp. Il carattere letto viene rigenerato al successivo mastro M_7
M_4 si staticizza nei FF Up il carattere letto in Mp
M_4
M_4 } vedi p.d.c. P_4
M_4
M_7 si scrive il risultato nel registro 0, decade delle centinaia

p.d.c. P_6

M_0 l'indirizzatore V conta +1
M_0 l'indirizzatore W conta —1
M_0 si ripone il FF Zc
M_0 scompare il segnale pMp e compare il segnale pMd

M_3 si ripone il FF Zr
M_3 si ripongono i FF Ud di uscita da Md
M_4 si legge in Md. Il carattere letto viene rigenerato al successivo mastro M_7
M_4 si staticizza nei FF Ud il carattere letto in Md
M_4 si legge nel registro T. Il carattere letto viene rigenerato
M_4 si staticizza nei FF Au il carattere letto in T
M_4 il carattere letto in memoria o nel registro T nel p.d.c. 6 sarà in generale un carattere alfanumerico i cui bits e ed f indicano se il numero in esame è compreso nella 1ª, 2ª, 3ª o 4ª decina di migliaia.

I bits e ed f non vengono operati nella UAD, ma in una apposita rete logica che tiene conto del codice degli indirizzi di memoria.

In M_4 vengono sommati i caratteri letti in memoria e in registro T. I bits a, b, c. d vengono operati in UAD, i bits e ed f nella rete logica a cui si è accennato; tale rete logica sarà esaminata in seguito.

M_7 si scrive il risultato nel registro 0, decade delle migliaia.

Nell'operare in UAD il quarto carattere dell'indirizzo IIII non viene trasferito in Kr l'eventuale riporto, ma di esso si tiene conto nella rete logica dei bits e ed f aumentando l'indirizzo di 10.000.

In questo modo l'operazione sul successivo carattere, relativa alla modifica di lunghezza, non viene influenzata dalla modifica di indirizzo.

p.d.c. P_7

M_0 l'indirizzatore V conta +1
M_0 l'indirizzatore W conta —1 (IWD in IWP)
M_0 si dispone il FF Zc
M_0 scompare il segnale pMd e compare il segnale pMp
M_3 si dispone il FF Zr
M_3 si ripongono i FF Up di uscita da Mp
M_4 si legge in Mp. Il carattere letto viene rigenerato al successivo mastro M_7
M_4 si staticizza in FF Up il carattere letto in Mp

M_4 si legge nel registro T. Il carattere letto viene rigenerato
M_4 si staticizza nei FF Au il carattere letto in T
M_4 si esegue in UAD la somma dei due caratteri letti.

Essi passano inalterati attraverso Co e Ca per la presenza dei segnali $\overline{IDM}$ $\overline{IDA}$

M_8 il risultato viene trascritto nel registro di lunghezza con $\alpha_L.P_7.\overline{\Delta T}$; α_L è un FF che in fase α è set ad ogni M_8 ed è reset ad ogni M_9
M_9 si ripone il FFα_{37} e scompare il segnale PT_I o PT_{II}

p.d.c. P_8

M_0 si azzera l'indirizzatore V
M_0 l'indirizzatore W conta —1 (IMP in IWD)
M_0 si ripone il FF Zc
M_0 scompare il segnale pMp e compare il segnale pMd
M_3 si ripone il FF Zr
M_3 si ripongono i FF Ud di uscita Md
M_4 si legge in Md. Il carattere letto viene rigenerato al successivo mastro M_7
M_4 si staticizza nei FF Ud il carattere letto in Md
M_4 si legge nel registro P l'indirizzo della DA e lo si pone in V riscrivendolo in P

E' selezionato P con $\alpha.P_8.\Delta A.\ \overline{\Delta AUR}.\ \overline{\Delta AEB}$; sono presenti: CJZ; CJV $= \alpha.P_8$; CVJ $= \alpha.P_8.\Delta A$

M_8 si scrive nel registro di lunghezza decine Ld il contenuto dei FF Ud con P_8.XLL; XLL $= \alpha_L.\overline{\Delta T}$
M_8 si ripone il FF KZ
M_9 si azzera il contatore di periodi Cp
M_9 si dispone il FF Po
M_9 si ripone il FF α
M_9 si dispone il FF β w

Con il riporsi del FF α ha termine la fase α di preparazione. Al termine della fase α si sono immagazzinati negli organi di governo I, O, W, V i seguenti indirizzi:

— l'indirizzo del primo carattere dell'istruzione successiva diminuito di 9 nel registro I (o nei registri M, H se la istruzione MA è stata letta rispettivamente sul 2° o sul 3° programma)

— l'indirizzo dell'operando nel registro 0

— l'indirizzo del carattere operato nel p.d.c. 8 della fase α nell'indirizzatore W

— l'indirizzo di inizio dell'accumulatore (stabilito dall'ultima DA) nell'indirizzatore V.

2.3. La fase β

Con l'attivarsi del FF β w ha inizio la fase esecutiva della istruzione MA.

Sequenza delle operazioni che seguono l'attivarsi del FFβw

Le operazioni della fase β differiscono tra loro a seconda che la prima cifra estratta da memoria sia un numero od un segno, e in relazione al modo in cui è stata espressa la lunghezza dell'operando.

La sequenza delle operazioni descritte si riferisce al caso in cui il 1° carattere estratto da memoria non sia un segno e in cui la sua lunghezza è espressa da due numeri introdotti in Lu Ld.

Alle operazioni relative agli altri casi si accenna alle « Note » relative ai periodi di cifra, in esse specificati.

p.d.c. P_0

M_0 il FF β w è disposto

M_0 si azzera l'indirizzatore W con $\beta w.\overline{\mu}.\overline{XWL}$
$\overline{XWL}$ è presente sicuramente

M_3 si ripone il FF α r

M_3 si dispone il FF β r

M_4 si trasferisce all'indirizzatore W il contenuto del registro 0 e si predispone il FF Zc (es.: $\overline{\Delta Ce}.IWD$).

Si seleziona 0 con $\beta w.P_0.\overline{XWL}$ (vedi M_0); è presente CJZ; è presente $CJW = \beta w.\overline{\eta}.\overline{\mu}.P_0$

M_5 si dispone il FF KZ

M_7 l'indirizzo letto in M_4 nel registro 0 viene riscritto nello stesso registro. $CWJ = \beta w.P_0.\overline{XWL}.\overline{\mu}$

M_9 il contatore Cp conta +1 con $KZ.\overline{\beta\Delta AEB}.2P_0\ \overline{Ar}$. Il decodificatore di p-pulsi fornisce il segnale P_1.

p.d.c. P_1

M_0 l'indirizzatore W conta —1 (IWD in IWP)

M_0 si dispone il FF Zc con $\overline{\Delta Ce}.IWD$

M_0 è presente il segnale pMp di consenso al pilotaggio di Mp
$PMp = \overline{XMY}.\overline{\varnothing d}.\overline{\eta}.XMX.XMC$
$XMC = \overline{A_r}.Z_c.P_o$
$XMX = \beta_r.\overline{\Delta AEB}$ (cioè non nella CM)
XMY = può essere presente in β solo nella ΔAER (FAM +FTM) o se in W unità c'è # (ISW presente)

M_0 è presente il segnale PA di consenso al pilotaggio di accumulatore con $\Delta A.\overline{A_s}.\overline{\Delta B}.\overline{\Delta AR}.\beta r.\overline{Po}.\overline{Ar}.\overline{\eta}$

M_3 si dispone il FF Zr con Zc

M_3 si ripongono i FF Up di uscita da Mp

M_4 si legge in Mp. Il carattere letto viene rigenerato con $CMpR = XMG.Zc$; $XMG = \beta r.\overline{\Delta V}.\overline{\Delta \Sigma B}.\overline{\Delta E}.\overline{\Delta MM}$

M_4 si staticizza nei FF Up il carattere letto in Mp

M_4 si legge in accumulatore all'indirizzo specificato dall'indirizzatore V, cancellandovi l'eventuale carattere contenuto. Il carattere letto non viene estratto; manca lo strobe di accumulatore essendovi $\beta w.\Delta U.\overline{\Delta B}$

M_7 si scrive in accumulatore il carattere staticizzato in Up.

Esso passa inalterato attraverso il complementatore Co per la presenza del segnale $\overline{IDM}$.

Passa inalterato anche attraverso l'UAD essendo Rm=1 ed essendoci zero all'ingresso BCa: infatti, per mancanza dello strobe l'uscita di accumulatore è 0 e passa inalterata attraverso il Ca essendoci $\overline{IDA}$.

Nota p.d.c. P_1

Se il primo carattere uscente da memoria è un segno, si dispone al mastro M_4 del p.d.c. P_1 il FF di segno da memoria Sm e viene a mancare il segnale CMU che consente il passaggio del carattere letto attraverso il complementatore Co. Al mastro M_9 del p.d.c. P_1 si dispongono il FF As (con Sm.ΔA.ΔU.ΔM) e il FF Sa (con $Sm.d_m.P_1.\Delta A.\Delta U.\Delta M$) nel caso di segno negativo o il solo FF As nel caso di segno positivo. Essi indicano il segno del numero introdotto in accumulatore.

Nella prima posizione dell'accumulatore si scrive ciò che esce dall'UAD, cioè tutti bit zero, a cui si aggiunge il K tramite l'and $\Delta A.\Delta UM.\overline{\eta}.P_1R.Sm$; tale aggiunta del K è necessaria per evitare una segnalazione di errore nelle istruzioni seguenti una MA di lunghezza 1 che trasferisce da memoria proprio un segno.

p.d.c. P_2

M_0	l'indirizzatore V conta +1 con $\Delta A.\beta w.\overline{Sm}.\overline{\Delta E}$
M_0	l'indirizzatore W conta —1 (IWP in IWD)
M_0	scompare il segnale pMp e compare il segnale pMd $PMd = XMY.\overline{\varnothing d}.\overline{\eta}.XMX.XME$; $XME = \overline{Ar}.\overline{Zc}.\overline{Po}$; per gli altri segnali vedasi P_1
M_0	si ripone il FF Zc con $\overline{\Delta Ce}.IWP$
M_3	si ripone il FF Zr con $\overline{Zc}$
M_3	si ripongono i FF Ud di uscita da Md
M_4	si legge in Md. Il carattere letto viene rigenerato: $CMdR = XMG.\overline{Zc}$
M_4	si staticizza nei FF Ud il carattere letto in Md
M_4	si legge in accumulatore all'indirizzo iniziale +1 cancellandovi l'eventuale carattere contenuto che non viene estratto
M_7	si scrive in accumulatore il carattere staticizzato in Ud. Esso passa inalterato attraverso il complementatore Co per la presenza del segnale $\overline{IDM}$.

Nota p.d.c. P_2

Se il primo carattere estratto è un segno, l'indirizzatore V non conta essendovi Sm. Esso ritarda di un p.d.c. ad iniziare la sua conta dato che il segno estratto da memoria non viene scritto in accumulatore. L'indirizzo iniziale rimane disponibile per il secondo p.d.c.

Ultimo p.d.c. pn

Nei periodi di cifra successivi al secondo, si ripetono le stesse operazioni.

Ad ogni p.d.c. il contatore di periodi Cp conta +1 e contemporaneamente il suo valore viene a confronto con il contenuto del registro L nel confrontatore di lunghezza C l. L'ultimo p.d.c. pn è quello in cui il contenuto di L eguaglia quello di Cp.

In questo caso si genera in Cl un segnale che al mastro M_5 dispone il FF ∅ cc e genera il segnale (∅c) di fine operando da memoria.

M_0 l'indirizzatore V conta +1

M_0 l'indirizzatore W conta —1

M_0 scompare il segnale pMp e compare il segnale pMd (o viceversa)

M_0 si ripone il FF Zc

M_3 si ripone il FF Zr

M_3 si ripongono i FF Ud di uscita di Md

M_4 si legge in Md. Il carattere letto viene rigenerato

M_4 si staticizza nei FF Ud il carattere letto in Md

M_4 si legge in accumulatore all'indirizzo specificato dell'indirizzatore V cancellandovi l'eventuale carattere contenuto che non viene estratto

M_5 si dispone il FF ∅ cc

M_7 si scrive in accumulatore il carattere staticizzato in Ud.

Esso passa inalterato attraverso il complementatore Co per la presenza del segnale $\overline{IDM}$

M_7 si scrive in accumulatore il bit gA di fine operando con $\overline{\eta}.\Delta U.\emptyset c.\overline{\Delta B}$

M_7 si dispone il FF φ con $\Delta M.\Delta U.\overline{\Delta V}.\emptyset c.\ (\overline{I^{\chi}} + \overline{C_{LU}})$; per quanto riguarda i segnali $\overline{I\chi}$ e $\overline{cLu}$ si veda il punto 2.4.

M_7 si ripone KZ su $\overline{KZ}$ con φ; si ripone ∅cc con φ

M_9 si ripone il FF βw con φ

M_9 si dispone il FF α

M_9 si azzera il contatore Cp con $\overline{KZ}$

M_9 si dispone il FF P_0 con $\overline{KZ}$

L'Unità Centrale è pronta per la lettura della istruzione successiva.

2.4. La fine su carattere chiave

La fine dell'operando da memoria può essere individuata oltre che ponendo in Lu Ld due numeri, nei seguenti modi:

1) Ponendo in Lu i caratteri Q o P e in Ld un carattere chiave. In questo caso il solo carattere chiave posto in Ld viene confrontato con i bits uscenti da memoria nel confrontatore Cr.

   Quando da memoria esce il carattere chiave si ottiene dal comparatore di ricerca il segnale I∅I. Tale segnale genera il se-

gnale $\emptyset$c tramite l'and $I\varkappa.\overline{\Delta RI}.\overline{P_0}.I\emptyset I.eLu$; $I\varkappa$ indica che in Lu c'è Q o P o Z o Y; eLu restringe la scelta a Q o P.

2) Ponendo in Lu il carattere Z o Y, e in Ld un carattere qualsiasi che non viene interessato alle operazioni. In questo caso si ottiene il segnale $Is\alpha$ appena si legge in memoria un carattere non numerico (questo caso è quindi riservato al trasferimento di soli caratteri numerici). Tale segnale genera il segnale $\emptyset$c tramite l'and $I\varkappa.\overline{\Delta RI}.IS\alpha.\overline{eLu}$; $\overline{eLu}$ restringe appunto la scelta a Z o Y.

La presenza di due diversi caratteri Q o P nel caso 1) e Z o Y nel caso 2) è giustificata dal fatto che talvolta si vuole trascrivere nel registro T_9 l'indirizzo dell'ultimo carattere dell'operando. Ciò è possibile nei casi in cui in Lu si pongono i caratteri Q o Z, contraddistinti dall'avere presente il bit « c » di codice.

In P o Y invece il « c » è assente.

Se in Lu sono stati posti i caratteri P o Y il segnale $\emptyset$c può disporre al mastro M_7 il $FF\varphi$ (essendo presente il segnale $\overline{cLu}$) e determinare la fine della fase βw. Se invece in Lu sono stati posti i caratteri Q o Z il segnale $\emptyset$c non attiva il $FF\varphi$ al mastro M_7 per la assenza del segnale $\overline{cLu}$.

Si susseguono quindi le seguenti operazioni:

- M_8 si ripone il FF KZ tramite il segnale $XFQ.\overline{\eta}$; $XFQ = I\varkappa.cLu.\overline{\Delta RI}.XFH$; XFH è il consenso al set del $FF\varphi$ dato da $\Delta M.\Delta U.\overline{\Delta V}.\emptyset c$
- M_9 si azzera il Cp e si dispone il FF P_0 con $\overline{KZ}$
- M_9 si dispone il FF η con $XFQ.\overline{\eta}_r$
- M_9 si ripone il FF IW con η
- M_9 si dispone il FF $\emptyset$d con $\emptyset$c.

La presenza dei segnali η o $\emptyset$d toglie il pilotaggio alla memoria. La fase β è terminata, e il contatore Cp è pronto per una nuova conta a partire da zero.

Si deve tenere presente che nel p.d.c. pn in accumulatore non viene trasferito il carattere chiave, bensì uno zero. Ciò è ottenuto bloccando l'uscita da Co per assenza del segnale CMU. Tale segnale è assente essendo presenti i segnali $I\varkappa.eLu.I\emptyset I$ (caso P o Q) o $I\varkappa.\overline{eLu}.IS\alpha$

(caso Z o Y). Ovviamente per avere in accumulatore uno zero con bit di controllo corretto si deve generare il bit k; ciò è ottenuto col segnale $\Delta U.\Delta M.\beta r.XM\alpha$; $XM\alpha = I\chi.eLu.I\varnothing I + I\chi.\overline{eLu}.IS\alpha$.

2.5. Fase η (sovrapposta alla fase β)

Durante tale fase si trasferisce nel registro T_9 l'indirizzo contenuto nel contatore W. L'indirizzo contenuto nel contatore W è quello in cui è stato trovato il carattere chiave che ha determinato la fine del trasferimento.

Si deve qui far notare che in realtà si trasferisce nel T_9 tale indirizzo incrementato di una unità. Ciò è dovuto al fatto che gli indirizzi specificati dal programma, sia indirizzi di istruzioni che di operandi, corrispondono sempre per la logica interna del calcolatore ad un indirizzo ridotto di una unità; quindi nel caso presente, passando da un indirizzo ottenuto dalla logica interna ad un indirizzo fornito al programma, si deve incrementare di uno l'indirizzo esistente nel W.

Conviene qui spendere qualche parola sul motivo per cui gli indirizzi nella logica interna del calcolatore sono sempre di una unità inferiori a quelli specificati nel programma.

Ricordiamo che, avendosi due canali di accesso alla memoria, sia in lettura che in scrittura, e avendosi due matrici di memoria, una pari ed una dispari, deve esistere un organo, e precisamente il FF Zc, che dica in ogni periodo di cifra quale matrice è collegata al canale interno. Il disporsi o il riporsi del FF Zc è condizionato allo stato pari o dispari dell'indirizzatore W.

Si può notare che il FF Zc è disposto dal segnale IWD (cioè indirizzo contenuto nel W dispari) al mastro M_0; ma allo stesso mastro M_0 nel W si ha una conta, e quindi l'indirizzo del W diventa pari; il FF Zc segnala infatti, quando è disposto, che la matrice pari deve essere collegata al canale interno.

La condizione di set di Zc è quindi valida perchè il W riceva sicuramente un impulso di conta, e ciò deve sempre avvenire all'M_0 di ogni p.d.c. in cui esista la necessità di pilotare la memoria principale; deve quindi avvenire anche all'M_0 del p.d.c. P_1; il primo carattere quindi verrà letto all'indirizzo specificato nell'istruzione diminuito di una unità.

Per evitare tale funzionamento occorrerebbe ritardare il disporsi o il riporsi di Zc rispetto alla conta del W, in modo da tener conto della futura posizione dell'indirizzatore W, che potrebbe avere contato (nei p.d.c. generici) o non contato (nel p.d.c. P_1); ciò ritarderebbe i pilotaggi di memoria e in definitiva si dovrebbe allungare il tempo del p.d.c.

Ovviamente di tale funzionamento si tiene conto in tutti i casi necessari, facendo sì che sempre ad un indirizzo specificato dal programma corrisponda per la logica interna un indirizzo diminuito di una unità.

p.d.c. P_0) Con segnale P_0 l'indirizzatore W non conta

- M_0 il FF Iw è riposto
- M_0 l'indirizzatore V si azzera con XWQ = $Ix.cLu.\overline{\Delta RI}.\eta.P_0$
- M_1 si scrive in V (decine) la cifra 9 per individuare il registro T_9 (con XWQ)
- M_3 si ripongono i FF Ud
- M_3 si ripone il FF Db con $PoR.\eta$

p.d.c. P_1) La conta dell'indirizzatore V è bloccata; compare il segnale CMU e quindi $\overline{IDM}$

- M_0 l'indirizzatore W conta + 1
- M_0 si dispone il FF Zc
- M_3 si dispone il FF Zr
- M_3 si ripongono i FF Rm
- M_3 è presente il segnale PT_I con XAF = $\eta.\beta w$
- M_4 si trasferisce nei FF Up la cifra unità dell'indirizzo posto in W; tale cifra, in and con P_1, genera i bit BWUa. Il set dei FF Up è dato dal consenso $I\eta Zc = \eta.Zc$.
- M_4 si legge nel registro T_9 cancellandone il contenuto. Il valore letto non viene estratto essendo inibito lo strobe da $\eta.\overline{\Delta S\Sigma R}$
- M_5 i FF Rm si predispongono sul valore 1
- M_7 si scrive nel registro T_9 il valore staticizzato in Up all'M_4 passando attraverso U.A.D. All'uscita dell'U.A.D. viene generato il bit K di controllo che non esiste nel W e viene anch'esso scritto nel T_9

p.d.c. P_2) l'indirizzatore W non conta

M_0 l'indirizzatore V conta + 1 con $\eta.\overline{P_1}$

M_0 si ripone il FF Zc

M_3 si ripone il FF Zr

M_3 si ripongono i FF Ud

M_4 si trasferisce in Ud la cifra decine dell'indirizzatore W.
Il set dei FF Ud è dato dal consenso $I\eta\overline{Zc} = \eta.\overline{Zc}$

M_4 si legge nel registro T_9 cancellandone il contenuto.
Il carattere letto non viene estratto

M_7 si scrive nel registro T_9 il carattere staticizzato in Ud al mastro M_4

p.d.c. P_3) l'indirizzatore W non conta

M_0 l'indirizzatore V conta + 1

M_0 si ripone il FF Zc

M_3 si ripone il FF Zr

M_3 si ripongono i FF Ud

M_4 si trasferisce nei FF Ud la cifra centinaia dell'indirizzatore W

M_4 si legge nel registro T_9 cancellandone il contenuto.
Il carattere letto non viene estratto

M_7 si scrive nel registro T_9 il carattere staticizzato in Ud al mastro M_4

p.d.c. P_4) l'indirizzatore W non conta

M_0 l'indirizzatore V conta + 1

M_0 si ripone il FF Zc

M_3 si ripone il FF Zr

M_3 si cancellano i FF Ud

M_4 si trasferisce nei FF Ud la cifra migliaia dell'indirizzatore W

M_4 si legge nel registro T_9 cancellandone il contenuto.
Il carattere letto non viene estratto

M_7 si dispone il FF Ga con $\eta.P_4$

M_7 si scrive nel registro T_9 il carattere staticizzato in Ud al mastro M_4 e si aggiunge il bit di fermo gA essendo disposto il FF Ga

M_7 si dispone il FF φ con $\eta . P_4$
M_7 si ripone il FF $\emptyset$ cc con φ
M_7 si ripone il FF Kz
M_9 si azzera il contatore Cp
M_9 si ripone il FF η
M_9 si dispone il FF po
M_9 si ripone il FF β w
M_9 si dispone il FF α

L'Unità Centrale è pronta per la successiva istruzione.

3. ISTRUZIONI DI TRASFERIMENTO MEMORIA - ACCUMULATORE

3.1. Considerazioni generali

Oltre alla MA ci sono altre due istruzioni di trasferimento interessanti la memoria e l'accumulatore. Esse sono:

a) Istruzione AM Decodificato ΔAEM
b) Istruzione A_0M Decodificato ΔAEP

Con tali istruzioni si trasferisce il contenuto dall'accumulatore dall'indirizzo posto dall'ultima DA per una lunghezza specificata nell'istruzione, all'indirizzo di memoria dato dall'istruzione stessa.
L'eventuale segno dell'accumulatore è pure trasferito in memoria e se ne tiene conto nella lunghezza.

La fine su carattere chiave si ha quando il carattere chiave viene trovato in memoria (e non in accumulatore); il carattere chiave è riscritto in memoria.

Nella AM il contenuto dell'accumulatore viene riscritto nello stesso; nella A_0M viene sostituito con zeri dotati di bit di fermo.
La fase di preparazione α di tali istruzioni è quasi identica a quella della MA; l'unica differenza è la mancanza del reset dei FF As, Sa e Cn.

3.2. La fase β

Vengono qui specificate solo le differenze rispetto alla MA. Si suppone che l'accumulatore non sia segnato e che la lunghezza sia espressa mediante due numeri. Opportune note chiariranno gli altri casi.

p.d.c. P_1

Non si ha rigenerazione della memoria (assenza XMG)
L'uscita dell'UAD non è usata (assenza di CUA).

Le uscite di accumulatore passano invariate attraverso il Ca per la presenza del segnale $\overline{IDA}$ e vengono scritte nella memoria tramite l'and XMM.ΔE.Zc (o $\overline{Zc}$); XMM = $\overline{XM\alpha}.\beta_r.\overline{\Delta S}.\overline{\Delta R}.\overline{\Delta B}$.

Nella AM esiste il comando di rigenerazione accumulatore (CAR = β_r.ΔE.ΔM).

Nella A_0M non vi è tale comando e si ha invece la rigenerazione del bit K collegato al codice dello zero tramite il segnale $\Delta E.\Delta P.\beta_r$; inoltre si ha il set del FF Ga di scrittura del bit di fermo in accumulatore (col segnale $\Delta P.\Delta E.XA\alpha.M_7$; dove $XA\alpha = \beta_r.\overline{P_0}.\overline{\eta}.\overline{\Delta RI}.PAT$).

Nota p.d.c. P_1

Se l'accumulatore è segnato manca il relativo pilotaggio (assenza di As); l'uscita di accumulatore (zero bit) si presenta all'ingresso della memoria; in or ad essa vengono posti i bit corrispondenti al segno dell'accumulatore + (IA+) o — (IA—) tramite il segnale $\Delta A.\Delta E.As.(\Delta P+\Delta M).\beta_r.P_1R$.

Da notare che l'effettivo segno dell'accumulatore è dato dalla combinazione del FF Sa (che ne indica il segno) e del FF Cn (che ne indica la rappresentazione in forma normale o in complemento). Quindi si ha:

$IA+ = \overline{Cn}.\overline{Sa} + Cn.Sa$; $\quad IA- = \overline{Cn}.Sa + Cn.\overline{Sa}$.

p.d.c. P_2

Come P_1; in più l'indirizzatore V conta + 1 (col segnale $\beta w.\Delta E.\overline{\Delta AEB}.\overline{\Delta RI}.\overline{As}$).

Nota p.d.c. P_2

Se l'accumulatore è segnato non avviene la conta + 1 dell'indirizzatore V. Inoltre se l'accumulatore è segnato ed in complemento si ha complementazione della cifra uscente dall'accumulatore, onde ottenere il risultato in memoria in forma normale. E' $DA = XAZ.\overline{\Delta AEB}.\Delta E.As.\overline{P_1}$; $XAZ = XAQ\ (\overline{\emptyset a} + gAu)$; $XAQ = \Delta A.\beta_r.Cn.\overline{\eta_r}$.
Si avrà quindi complementazione solo finchè arriverà il bit di fermo di accumulatore.

Per la prima cifra uscente in P_2 e per le successive, fino alla prima diversa da zero compresa si ha complementazione a 10, cioè esiste anche il comando + 1.

Ultimo p.d.c. pn

Nei periodi di cifra successivi al secondo si ripetono le stesse operazioni; l'ultimo p.d.c. pn è quello in cui il contenuto di L eguaglia

quello di Cp. Si segue quindi la stessa procedura di fine della MA. Il consenso al disporsi del FF φ è dato dal segnale: $(\overline{I_X} + \overline{cLu}).XFH$. Dove il segnale XFH è equivalente a: $\Delta E . \overline{\Delta AEB} . \overline{\Delta V} . \emptyset c$.

Nota p.d.c. pn

La fine dell'operando può, come è stato accennato, essere individuata da un carattere chiave posto in memoria. Tale carattere chiave deve essere riscritto in memoria, al posto del carattere in arrivo da accumulatore.

Si ha il segnale $XM\alpha$ (vedasi MA) che indica appunto che è stato trovato in memoria il carattere chiave; tale segnale impedisce che si scriva in memoria ciò che arriva dall'accumulatore (vedasi in P_l il segnale XMM) e abilita il comando di riscrittura in memoria XMG $= \overline{\Delta AER}.\beta_r.\overline{\Delta V}.XM\alpha$.

Per quanto riguarda i vari casi di fine su carattere chiave vale quanto si è detto per la MA.

4. ISTRUZIONI DI TRASFERIMENTO MEMORIA-REGISTRI T

4.1. Considerazioni generali

Le istruzioni di trasferimento fra Memoria principale e registri ausiliari T sono:

a) Istruzione MT decodificato ΔTUM

b) Istruzione TM decodificato ΔTEM

c) Istruzione T_0M decodificato ΔTEP

d) Istruzione Y decodificato ΔSUM

Le tre istruzioni di cui ai punti a, b, c si presentano come successione di 8 caratteri alfanumerici: L T_o IIII T_m F che vengono letti da destra a sinistra.

F, T_m, IIII hanno la funzione già nota.

T_o è il carattere indicante l'indirizzo del registro interessato dal trasferimento.

L è la lunghezza dell'operando, non superiore a 10 dato che tale è la massima lunghezza di un registro.

4.2. La fase α

Per i periodi di cifra da P_0 a P_5 l'unica differenza rispetto alla MA è che non si ha il reset dei FF As ed Sa.

p.d.c. P_7

La somma dei due caratteri letti in memoria e nel registro modificatore non viene scritta nel registro di lunghezza (assenza di $\overline{\Delta T}$).

p.d.c. P_8

Non si trasferisce nell'indirizzatore V il contenuto del registro JP (assenza di ΔA), ma si trasferisce in tale indirizzatore il risultato ottenuto in P_7 dalla somma dei due caratteri letti in memoria e nel registro modificatore; ciò si ottiene col segnale XWC.M_1 - dove XWC $= P_8 . \Delta T . \alpha$.

L'individuazione del registro corrispondente al risultato ottenuto è effettuata colla stessa rete logica descritta nella MA al p.d.c. P_2 della fase α.

Il carattere L letto in memoria viene scritto nel registro lunghezza unità, invece che nelle decine (per la presenza di ΔT).

4.3. La fase β

E' analoga a quella della corrispondente istruzione relativa all'accumulatore (cioè la MT corrispondente alla MA, la TM alla AM, ecc.). L'unica differenza è che esiste il comando di pilotaggio per i registri T, cioè il segnale $XAF = \Delta T.\overline{\Delta B}.\beta_r.\overline{\Delta AER}$.

Inoltre: l'eventuale segno uscente da memoria nella MT non influisce sulla conta dell'indirizzatore V, e quindi viene scritto nella prima posizione del registro; non esistono indicatori di registri segnati o in complemento, e quindi nelle TM, T_0M non si può avere nè generazione del segno nè complementazione.

La lunghezza dell'operando può anche essere determinata con fine su carattere chiave, ma ovviamente solo per i casi Z o Y (cioè fine su qualsiasi carattere non numerico).

4.4. L'istruzione Y

Tale istruzione si presenta come una successione di 8 caratteri alfanumerici LL IIII T F aventi lo stesso significato della MA.

Come è noto l'istruzione Y trasferisce LL caratteri dalla memoria al registro T_0 ponendo l'eventuale segno nell'apposito registro Sr.

La fase α è analoga alla MA; le differenze sono le seguenti:

a) non si ha trasferimento dell'indirizzo di JP in V al p.d.c. P_8 (assenza di ΔA); però essendo azzerato V in $P_8.M_0$ si è automaticamente ottenuto l'indirizzo del T_0.
b) In $\alpha.P_2.\Delta S.\Delta U.\Delta M.M_6$ si ripone il FF Sr.

La fase β presenta le seguenti differenze:

a) la conta dell'indirizzatore V è data ad ogni M_0 dal segnale $\beta w.\overline{Sm}.\Delta S.\Delta U.\Delta M$.

   Non si ha quindi conta in $P_2.M_0$ se da memoria è uscito un segno in P_1.

b) L'eventuale segno negativo dispone il FF Sr col segnale: $\Delta S.\Delta U.\Delta M.\beta_w.P_1.Sm.d_{MI}.M_9$.
c) Si ha il pilotaggio ai registri tramite il segnale: $XAF = \Delta S.\Delta U.\Delta M.\beta_r$.

Anche qui si può avere la fine su carattere chiave con comportamento identico alla istruzione MA.

5. ISTRUZIONI DI TRASFERIMENTO MEMORIA-MEMORIA

5.1. Istruzione PUM (LL IIII T ÷)

Decodificato ΔSΣB; ΔPI

La PUM è istruzione preparatoria per le istruzioni: MEM, CMM, + MM, — MM che essa precede nel programma.

La PUM prepara l'uscita da Memoria dei caratteri che verranno operati nelle istruzioni precedentemente citate; infatti dall'indirizzo di Memoria IIII modificato da T dell'istruzione PUM, trasferito durante la PUM stessa nell'indirizzatore Y, si opera successivamente come segue:

MEM : si prelevano i caratteri nei trasferimenti da una zona di Memoria ad un'altra.

+ MM: si prelevano i caratteri nelle somme fra numeri contenuti in diverse zone di Memoria.

— MM: si parte a considerare i caratteri dai quali vengono sottratti quelli contenuti in altra zona di Memoria.

CMM : si parte per ottenere la parola che costituisce il secondo termine di confronto con altra parola contenuta in Memoria.

La lunghezza dell'operando può essere espressa dalle cifre estratte ai p.d.c. p_7, p_8 della fase α della PUM.

In tal caso nella istruzione che la segue non verrà indicata la lunghezza dell'operando. Se invece la lunghezza dell'operando viene indicata nella istruzione che segue la PUM, questa non contiene la indicazione della lunghezza e ai p.d.c. p_7, p_8, si legge in Memoria il carattere #.

L'istruzione PUM staticizza nell'indirizzatore Y l'indirizzo individuato. Ciò è possibile solo se il FF βy è riposto, cioè se non è in corso di svolgimento una istruzione che impegni il canale esterno.

Se il canale esterno è impegnato l'indirizzo individuato dalla istruzione PUM non può essere introdotto nell'indirizzatore Y e l'istruzione ricicla finchè non si ripone il FF βy. Questo caso verrà esaminato a proposito dell'istruzione PIN del tutto analoga alla PUM.

5.2. **Preparazione ed esecuzione dell'istruzione PUM**

Fase α

Dura 8 p.d.c. e determina il trasferimento del carattere di funzione nei FF F, dell'indirizzo modificato nel registro 0, e della lunghezza dell'operando nei FF Lu, Ld.

Fase β

La fase β dura un solo p.d.c. (P_0)

- M_0 si azzera l'indirizzatore W
- M_0 si azzera l'indirizzatore Y (se è $\overline{\beta y}$)
- M_4 si trasferisce negli indirizzatori W e Y il contenuto del registro 0.
- M_7 si dispone il FF βy con $\beta w.\overline{\alpha_2}.\overline{\alpha_p}.\overline{Ni}.\Delta By$
- M_7 si dispone il FF φ con $\beta_w.P_0.\Delta PI$
- M_7 si ripone il FF Kz
- M_9 si azzera il contatore Cp.
- M_9 rimane disposto il FF P_0
- M_9 si dispone il FF α.

Disponendosi il FF α si passa a preparare l'istruzione successiva.

5.3 **Istruzione MEM** (00 IIII T_1)

Decodificato ΔSUP; ΔMM

Con la istruzione MEM si trasferisce una parola o un numero da una zona di Memoria con indirizzo iniziale specificato nella istruzione PUM che la precede, ad una zona di Memoria con indirizzo iniziale IIII modificato dal contenuto di T, espresso dalla stessa istruzione MEM.

La lunghezza delle due parole è uguale e può essere specificata indifferentemente nella PUM o MEM.

Contemporaneamente alla MEM non possono essere eseguite altre istruzioni poichè essa impegna i canali interno ed esterno.

5.4. **Preparazione della istruzione MEM**

Fase α

La fase α dura 8 p.d.c. e determina il trasferimento del carattere di funzione nei FF F, dell'indirizzo modificato nel registro 0, e della lunghezza dell'operando nei FF Lu, Ld. I FF Lu, Ld non vengono

azzerati al p.d.c. p_1 della fase α se la precedente istruzione PUM ha provveduto a definire la lunghezza dell'operando. In tale caso ai p.d.c. p_7, p_8 della fase α vengono letti in Memoria due caratteri 0 che non alterano il contenuto dei FF Lu, Ld. Se invece nella precedente istruzione PUM sono stati operati due caratteri # ai p.d.c. p_7, p_8, al mastro M_7 del p.d.c. p_1 della fase α dell'istruzione MEM si azzerano i FF Lu, Ld e ai successivi p.d.c. p_7, p_8 la lunghezza dell'operando può essere staticizzata in Lu, Ld; ciò è ottenuto col comando di azzeramento $\alpha.P_1.M_7.\Delta MM.\Delta\eta L_{10}$; $\Delta\eta L_{10}$ è il decodificato di # in L decine.

Durante la fase α il contatore W conta +10 al p.d.c. p_0 e —1 ai successivi p.d.c.; il contatore Y non conta poichè il FF Ce è riposto.

5.5. Esecuzione della istruzione MEM

Fase β

p.d.c. P_0 M_0 si azzera l'indirizzatore W

M_3 è presente il segnale XMJ che abilita l'ingresso nella seconda zona di Memoria dei caratteri letti nella prima zona di Memoria

M_3 si ripone il FF α_r e scompare il segnale del pilotaggio in Memoria

M_3 si dispone il FF βr

M_4 si ripone il FF I_y

M_4 si trasferisce all'indirizzatore W il contenuto del registro 0

M_5 si dispone il FF Kz

M_8 si dispone il FF Ce. Se supponiamo pari il numero contenuto in Y, si ha il segnale Ce.IYP

M_9 si ripone il FF P_0

M_9 il contatore Cp conta +1. Il decodificatore di p-pulsi fornisce il segnale P_1.

p.d.c. P_1 M_0 l'indirizzatore W conta —1

M_0 l'indirizzatore Y conta —1 col segnale $Ce.M_0.\overline{XW\lambda}$; $XW\lambda = ISY + \mu.P_0$ dove ISY è il decodificato del carattere # nell'indirizzatore Y unità (in modo analogo a ISW)

P_1 M_0 si dispone il FF Zc con Ce.IYP

M_0 sono presenti i segnali pMp, pMd di consenso al pilotaggio della Memoria Mp, Md.
$pMp = \overline{XMY} . \overline{\emptyset d} . \overline{\eta} . XMX.XMC.$ (vedasi MA)
$pMd = Ce . \beta y . Zc . \overline{XW\lambda}$

M_3 si dispone il FF Zr

M_4 si legge in Md e si staticizza il carattere letto nei FF Ud; dalle uscite di questi si ottengono i BME (Z_r).

Si legge in Mp e si staticizza il carattere letto nei FF Up; dalle uscite di questi si ottengono i BMI.

Il carattere staticizzato in Up non può transitare attraverso il complementatore Co per l'assenza del segnale CMU ($\Delta S.\Delta U.\Delta P$ inibisce l'uscita di CMU); il carattere staticizzato nei FF Ud (BME) viene prelevato attraverso sette inverter dalle uscite negative dei FF Ud e inserito in or sull'uscita del Ca tramite il consenso $\Delta MM.\overline{Am}$.

M_7 si scrive in Memoria Mp attraverso i sommatori S_1, S_2 e la porta XMJ.Zc il carattere letto in Memoria Md
$XMJ = \Delta MM.\overline{\Delta SEB}.\beta_r.\overline{XM\alpha}.$

M_7 si rigenera in Md il carattere letto (BME), essendo presente il segnale CMdR = XMF.Zc;
$XMF = Ce.\beta y.\Delta MM.\overline{PRN}$

M_9 il contatore Cp conta +1. Il decodificatore di p-pulsi fornisce il segnale P_2.

p.d.c. P_2 M_0 l'indirizzatore W conta —1

M_0 l'indirizzatore Y conta —1

M_0 si ripone il FF Zc

M_3 si ripone il FF Zr

M_4 si legge in Memoria Mp. Il carattere viene rigenerato col successivo mastro M_7. $CMpR = XMF.\overline{Zc}$

M_4 si legge in Memoria Md. Il carattere non viene rigenerato al mastro M_7

M_4 si staticizzano i caratteri letti in M_p e M_d nei FF Up, Ud

P_2 M_7 si scrive, nel modo già descritto al p.d.c. precedente, in Memoria Md il carattere letto al mastro M_4 in Memoria Mp

M_9 il contatore Cp conta +1. Il decodificatore di p-pulsi fornisce il segnale p_3.

Le operazioni descritte si ripetono nei successivi p.d.c. finchè non si ottiene dal comparatore di lunghezza il segnale che dispone al mastro M_5 dell'ultimo p.d.c. il FF ∅cc. Al mastro M_7 si dispone il FF φ che determina la fine dell'istruzione; XFH = ΔMM.∅c.

Questa può essere individuata oltre che ponendo due numeri in Lu, Ld, per mezzo dei caratteri Q, P o Z, Y posti nei FF Lu al p.d.c. p_7 della fase α.

In questo caso si ottiene la sequenza di operazioni già descritta nella istruzione MA. L'ultimo carattere operato (carattere chiave se in Lu è stato posto P o Q, carattere non numerico se in Lu è stato posto Z o Y) non viene trasferito dalla prima alla seconda zona di Memoria per assenza del segnale XMJ.

Infatti con i segnali I∅I e ISα che si generano con l'ultimo carattere operato, si ottiene il segnale XMα che impedisce il trasferimento alla seconda zona di Memoria del carattere letto nella prima zona e consente la riscrittura nella seconda zona del carattere lettovi; si ha infatti il segnale XMG.

Nel caso in cui all'inizio della fase β entrambi gli indirizzatori W e Y contengano indirizzi pari o dispari, si dispone in βw, p_0 M_8, il FF Ar e in βw, p_0, M_9 il FF Am. Contemporaneamente al FF Ar si dispone il FF PoC.

La presenza del segnale Ar che dura fino al mastro M_8 del successivo p.d.c., impedisce in βw, M_0 la conta dell'indirizzatore W, fa mancare il corrispondente pilotaggio della Memoria, e impedisce in βw, P_0, M_9 la conta del contatore Cp. Il FF P_0 viene ugualmente riposto. Si ha così un secondo p.d.c. caratterizzato dalla presenza del segnale PoC.

Nel p.d.c. PoC conta solo l'indirizzatore Y e si procede ad estrarre da Memoria il primo carattere che viene staticizzato nei FF Ud (Up)

di uscita Memoria e trasferito al mastro M_3 del p.d.c. successivo (P_1) ai FF Au di uscita accumulatore; (col consenso ΔMM.Am.$\bar{\eta}$).

Nei p.d.c. successivi si ha in M_0 la conta di entrambi gli indirizzatori, in M_3 il trasferimento nei FF Au del carattere letto al p.d.c. precedente, in M_7 il trasferimento in Memoria del contenuto dei FF Au. Il FF Am rimane disposto per tutta la fase βw della istruzione MEM. I FF Au vengono riposti in ogni p.d.c. con ΔMM.βr.M_2, appena prima che vi vengano staticizzati i FF Ud (Up) all'M_3.

6. ISTRUZIONI +MA e —MA

6.1. Istruzione +MA

Decodificato ΔAΣP; ΔK
Con la istruzione +MA si opera la somma di un numero estratto da Memoria a partire dall'indirizzo IIII modificato dal contenuto del registro T col numero contenuto in accumulatore a partire dall'indirizzo iniziale.

La lunghezza del numero in Memoria è indicata dalla istruzione stessa, quella del numero in accumulatore è definita dalla presenza del bit gA sull'ultimo carattere operato. Il risultato della somma viene posto in accumulatore a partire dall'indirizzo iniziale. Entrambi i numeri possono essere segnati; il risultato potrà essere espresso in forma normale o in complemento.

6.2. Preparazione ed esecuzione dell'istruzione +MA

Fase α

La fase α dell'istruzione +MA ha la durata di 9 p.d.c. (p_0-p_8), durante i quali si opera in modo del tutto analogo a quello descritto nell'istruzione MA, trasferendo l'indirizzo iniziale dell'operando da Memoria nel registro 0 e la sua lunghezza nei FF Lu, Ld. In α, p_8, M_9 si ripone il FF α e si dispone il FF βw che temporizza l'inizio della fase esecutiva β.

Fase β

Gli organi logici che intervengono nell'esecuzione della istruzione +MA sono: l'Unità Aritmetica, il FF Sm, il FF Da, il FF Db, i FF As, Sa, il FF Cn, oltre agli organi relativi alla estrazione dei dati da Memoria e da accumulatore.

Il significato dei FF testè citati è il seguente:

FF Sm : indica, disponendosi in βw, p_1 che il numero da Memoria è segnato

FF Da : indica, riponendosi in βw, M_2, che nel p.d.c. precedente è uscito da Memoria un carattere diverso da zero

FF Db : è inizialmente riposto e può successivamente disporsi e riporsi a seguito del segnale indicante che l'operando in

accumulatore ha segno negativo (in βw, p_0, M_0), e del segnale indicante che l'operando in Memoria ha segno negativo (in βw, p_2, M_0).

All'arrivo del primo carattere da Memoria che segue l'eventuale segno, il FF Db sarà quindi disposto solo se uno dei due operandi ha segno negativo. In questo caso tutti i caratteri uscenti da Memoria vengono complementati a dieci fino al primo carattere significativo compreso, e i successivi a nove. Se invece il FF Db rimarrà riposto i caratteri uscenti da Memoria non verranno complementati.

FF As ed Sa: indicano il segno dell'operando contenuto in accumulatore. Se il segno è positivo è disposto il solo FF As, se il segno è negativo sono disposti entrambi i FF As ed Sa. Lo stato di questi FF non viene ovviamente definito durante l'istruzione + MA, ma nella precedente istruzione MA con la quale si è trasferito uno degli operandi nell'accumulatore. Lo stato iniziale di questi FF non subisce mutamenti nel corso dell'istruzione + MA.

FF Cn : indica, disponendosi o meno, se il risultato in accumulatore è in forma normale o in complemento. Nel caso in cui l'operando da Memoria viene complementato, esso si dispone se, nelle operazioni relative all'ultimo carattere non si è ottenuta indicazione di riporto.

Passiamo ora a descrivere le principali operazioni della fase β, riferendoci al caso in cui il numero in accumulatore è positivo e quello in Memoria è negativo.

Fase β

p.d.c. p_0

M_0 si azzera l'indirizzatore W

M_4 si trasferisce all'indirizzatore W il contenuto del registro 0

M_5 si dispone il FF Kz

M_9 il contatore Cp conta + 1

p.d.c. p_1

M_0 l'indirizzatore W conta — 1

M_0 sono presenti i segnali pMp (pMd) di consenso al pilotaggio della memoria Mp (Md) e PA di consenso al pilotaggio dell'accumulatore. Si ripongono i FF Up (Ud) e i FF Au; $PA = \Delta A.\Delta K.\overline{\Delta B}.\overline{A_r}.\beta_r.\overline{Po}.\overline{\eta}.$

M_4 si estrae da Memoria il primo carattere, cioè il segno del numero contenuto in Memoria. Esso viene staticizzato nei FF Up (Ud), e rigenerato in Memoria al mastro M_7 successivo. Contemporaneamente ai FF Up (Ud) si dispone il FF Sm.

Il carattere uscito dall'accumulatore viene rigenerato con $CAR = \Delta A.\overline{\Delta E}.Sm.P_1R.$

p.d.c. p_2

M_0 l'indirizzatore W conta —1. Si dispone il FF Db (segno — da Memoria)

M_4 si estrae da Memoria la prima cifra dell'operando e la si staticizza nei FF Ud (Up); essa verrà riscritta in Memoria al mastro M_7. Si estrae dall'accumulatore la prima cifra dell'operando e la si staticizza nei FF Au; essa non verrà riscritta in accumulatore al mastro M_7 per assenza del segnale CAR.

Il carattere da Memoria complementato a 10 e il carattere da accumulatore, inalterato, vengono operati nell'unità aritmetica.

L'eventuale riporto non viene operato nello stesso p.d.c., ma conteggiato dal contatore Kr al mastro M_8

M_7 si scrive in accumulatore all'indirizzo del V il risultato della somma delle prime due cifre operate

M_9 il contatore Cp conta + 1.

p.d.c. p_3

M_0 l'indirizzatore W conta —1; l'indirizzatore V conta +1

M_3 si ripongono i FF Ud (Up) e i FF Au. Si azzera il registro dei riporti Rr. Si ripone il FF Da

M_4 si estrae da Memoria la seconda cifra e la si staticizza nei FF Up (Ud).

Si estrae da accumulatore la seconda cifra e la si staticizza nei FF Au.

I caratteri letti in Memoria e in accumulatore vengono introdotti nella U.A.D. attraverso i complementatori Co, Ca. Il carattere proveniente da Memoria viene complementato a 9, quello proveniente da accumulatore non viene complementato.

L'U.A.D. opera sulle cifre introdotte come al p.d.c. precedente

M_5 il contatore di riporti Kr trasferìsce il suo contenuto al registro Rr. Il registro Rr invia a sua volta il suo contenuto alla U.A.D.; il riporto ottenuto al p.d.c. precedente viene quindi sommato al risultato ottenuto al mastro M_4

M_7 si scrive in accumulatore il risultato della somma delle seconde due cifre

M_8 il contatore Kr tiene conto dell'eventuale riporto ottenuto in M_4, se c'è stato riporto in M_5, esso viene conteggiato da Kr al mastro M_1 del p.d.c. successivo.

Nei p.d.c. successivi si opera in modo del tutto eguale a quello descritto per il p.d.c. p_2 con conta avanti dell'indirizzatore V e conta indietro dell'indirizzatore W.

La fine dell'istruzione è data dalla presenza di entrambi i segnali $\emptyset$c $\emptyset$a che indicano rispettivamente la fine dell'operando da Memoria e di quello da accumulatore.

Se il numero proveniente da Memoria è più lungo di quello proveniente da accumulatore, l'uscita del segnale $\emptyset$a precede quella del segnale $\emptyset$c. Con la presenza del bit ga sull'ultimo carattere da accumulatore si dispone il FF $\emptyset$a con $gAu.\bar{p}.\overline{Sm}.M_6$. La presenza di $\emptyset$a inibisce lo strobe di accumulatore e dallo stesso si avrà quindi uscita zero bit, equivalente a zero codice a parte il bit K. Nel p.d.c. relativo all'ultimo carattere estratto da Memoria, sia per lunghezza specificata numericamente che con carattere chiave, si ottiene il segnale $\emptyset$c.

Se la fine avviene su carattere chiave, tale carattere non passa attraverso il Co per assenza del segnale CMU, e quindi ai fini dell'operazione aritmetica si ottiene uno zero da Memoria. Nel caso

in cui si debba avere complementazione a 9 si ha un apposito segnale ($XM\gamma = XM\alpha.dn.\overline{Db}.\overline{Da} + XM\alpha.\overline{dn}.Db.\overline{Da}$) che introduce i bit c e d (cioè un 9) all'uscita del Co.
Per la presenza di $\emptyset a\,.\,\emptyset c\,.\,\overline{Ar}.\,\overline{\Delta AX}.Db.\overline{Cn}$ si ottiene il segnale $\emptyset$Am di consenso al disporsi del FF φp. Il FF φp si dispone se non vi è fine su carattere chiave, oppure se, pur essendo tale, si ha P o Y in Lu, cioè non occorre effettuare la fase η.

Con le stesse condizioni il segnale $XFH = \emptyset Am.\Delta\Sigma.\overline{\Delta V}.\overline{\Delta S}$ dispone il FF φ.

Il segnale $\emptyset$Am provvede a scrivere il bit gA in accumulatore, disponendo il FF Ga con $\emptyset Am.\Delta K.\overline{\eta}.\overline{\Delta TX}.M_7$.

Le operazioni relative alla + MA terminano al mastro M_2 del p.d.c. successivo, in cui si dispone il FF Cn, se nell'ultimo p.d.c. non si è avuto riporto.

Se invece il numero proveniente da accumulatore è più lungo di quello proveniente da Memoria, dopo che è stato letto in Memoria l'ultimo carattere si ha il segnale $\emptyset$c che dispone all'M_9 il FF $\emptyset$d.

Tale segnale inibisce il pilotaggio di Memoria; dalla Memoria escono pertanto zero bit che attraverso il Co diventano 9. Quando si ha l'uscita del gA da accumulatore tutto avviene come nel caso precedente.

Per poter giudicare quando, all'apparire dei segnali $\emptyset$c e $\emptyset$a, è possibile finire immediatamente l'istruzione, senza estenderla di un p.d.c. per gli eventuali riporti, occorre analizzare tutti i casi possibili.

Questi sono dati da tutte le possibili combinazioni dei segnali Cn, Db, IRφ.

E' noto cosa significhino Cn e Db; IRφ indica che vi è un riporto in S_1 o in S_2.

Si avranno quindi 8 casi rappresentati nel seguente specchietto:

	Db		$\overline{Db}$	
Cn	fine immediata	1 p.d.c. prolungamento	fine immediata, riporre Cn	fine immediata
$\overline{Cn}$	fine immediata	fine immediata, disporre Cn	prolungamento 1 p.d.c.	fine immediata
	IRφ	$\overline{IR\varphi}$	IRφ	$\overline{IR\varphi}$

Ci si può rendere giustificazione di tale specchietto supponendo di avere sempre l'estensione di un p.d.c.; dal risultato che si otterrebbe si vede se tale prolungamento è necessario o meno.

Ad esempio: caso $Db.\overline{Cn}.\overline{IR\varphi}$. Se si prolungasse di 1 p.d.c. si avrebbe uno 0 dal Ca, un 9 dal Co, e non avendosi riporto si scriverebbe tale nove in accumulatore e così via per qualsiasi p.d.c.; è inutile continuare, occorre però disporre Cn.

Caso $\overline{Db}.\overline{Cn}.IR\varphi$; è ovvio che si deve ancora scrivere il riporto.

Quindi, quando si hanno i segnali $\emptyset a.\emptyset c.\overline{Ar}.\overline{\Delta AX}$ si può subito avere $\emptyset Am$, cioè la fine dell'operazione, nei casi:
$\overline{Db}.\overline{IR\varphi} + Db.\overline{Cn} + Db.IR\varphi + \overline{Db}.Cn$.

6.3. Istruzione — MA

Decodificato $\Delta A \Sigma M$; ΔK

E' perfettamente analoga alla + MA; l'unica differenza è costituita dal fatto che il FF Db, connesso a contatore, in βw, P_0, M_6 (col segnale $\overline{\eta}.\Delta M.P_0.M_6$) riceve un impulso di conta ulteriore, che tiene conto del segno negativo dell'operazione; ciò perchè appunto il FF Db, commutando ad ogni segno negativo, dà il segno risultante.

7. ISTRUZIONI + X e — X

7.1. Istruzione + X (LL IIII T =)

Decodificato: Δ A X P

Moltiplicazione additiva

Con l'istruzione + X si moltiplica un numero (moltiplicando), estratto da Memoria a partire dall'indirizzo IIII modificato dal contenuto di T, per un numero (moltiplicatore) posto nei registri T con la cifra meno significativa all'indirizzo 00; al risultato si aggiunge un numero addendo) contenuto in Accumulatore, estratto a partire da un qualsiasi indirizzo specificato da una precedente istruzione DA.

Il risultato complessivo viene scritto in Accumulatore a partire dall'indirizzo da cui si è iniziata l'estrazione del numero operato. L'esecuzione dell'istruzione + X consta di un numero n di cicli pari al numero delle cifre del moltiplicatore se la sua cifra più significativa è < 5, o di $n + 1$ cicli se essa è $\geqslant 5$.

Nel primo p.d.c. si opera anzitutto il trasferimento della prima cifra del moltiplicatore dai registri T ai FF Rm, poi si inizia la estrazione delle cifre del moltiplicando da Memoria e di quelle dell'addendo da Accumulatore. Nei singoli p.d.c. si esegue l'operazione M.Rm + A dove M è la cifra proveniente da Memoria, Rm è la cifra staticizzata nei FF Rm ed A è la cifra proveniente da Accumulatore.

Il risultato viene scritto in Accumulatore ad ogni p.d.c.

Nel ciclo successivo si cancella il contenuto di Rm e vi si trasferisce la seconda cifra del moltiplicatore, poi si inizia di nuovo l'estrazione delle cifre del moltiplicando da Memoria e del primo risultato parziale da Accumulatore, partendo però dall'indirizzo della seconda cifra di Accumulatore ed operando come già detto per il primo ciclo.

Nei cicli successivi si ripetono le stesse operazioni, spostandosi ogni volta di una cifra in avanti nell'estrarre il contenuto di Accumulatore. Il trasferimento di ogni cifra del moltiplicatore nei FF Rm è caratterizzato dalla fase ρ che segue immediatamente la fase α preparatoria. L'esecuzione delle operazioni relative ad ogni ciclo è caratterizzata dalla fase β che succede alla fase ρ.

La fine di un ciclo è indicata dal segnale φp che attiva di nuovo la fase ρ. Le fasi ρ e β si alternano fino a che non si dispone al termine dell'ultima fase β il FF φ.

Si hanno tanti cicli quanti sono i caratteri del moltiplicatore, aumentati di 1 se l'ultimo carattere del moltiplicatore è $\geqslant 5$.

La lunghezza del moltiplicando è indicata dal contenuto dei FF Lu, Ld, quella del moltiplicatore e dell'addendo dalla presenza del bit g sull'ultimo carattere operato.

Ciascuno dei tre numeri può essere segnato. Il risultato in Accumulatore può essere espresso in forma normale o in complemento.

L'istruzione + X viene preceduta da una istruzione Y e da una istruzione DA. Con la istruzione Y si introduce il moltiplicatore nei registri T a partire dall'indirizzo 00; se il segno è negativo si dispone il FF Sr. Con la istruzione DA si trasferisce nel registro JP in α, p_3 e p_4 l'indirizzo di Accumulatore da cui si inizierà ad operare nella fase β della successiva istruzione + X.

Nei registri Rm possono essere introdotte solo cifre < 5; se la cifra operata è $\geqslant 5$, essa viene complementata.

La eventuale complementazione avviene durante la fase ρ, in cui si trasferisce una cifra del moltiplicatore nei FF Rm. La complementazione avviene nel complementatore Ca. Se la cifra è stata complementata, si dispone il FF dn che è inizialmente riposto.

La logica dell'unità aritmetica richiede che la complementazione delle cifre del moltiplicatore avvenga come segue (vedasi anche descrizione logica degli organi; pag. 40 e segg.):

prima cifra	complementata a 10	se $\geqslant 5$
	invariata	se < 5
cifre successive alla 1ª	complementate a 10	se $\geqslant 5$ e FF dn riposto
	complementate a 9	se $\geqslant 5$ e FF dn disposto
	aumentate di 1	se < 5 e FF dn disposto
	invariate	se < 5 e FF dn riposto

Il numero introdotto in Accumulatore con una precedennte istruzione MA per essere operato successivamente nella istruzione + X. può essere segnato; se il segno è + si dispone con la MA il FF As, se il segno è — si dispongono i FF As, Sa. Nell'esecuzione della istru-

zione + X le cifre uscenti da Accumulatore non vengono complementate ad eccezione di quelle estratte dopo la fine dell'operando in Accumulatore (presenza del segnale ∅a). Per queste cifre (esse sono ovviamente degli zeri) la complementazione è possibile se il FF Cn è disposto.

Il FF Cn staticizza alcune condizioni relative all'ultimo carattere operato nel ciclo precedente a quello in esame; esso si dispone col segnale φp se l'ultimo carattere estratto da Memoria in quel ciclo, è stato complementato e non ha dato luogo a riporto.

Si ripone col segnale φp se l'ultimo carattere estratto da Memoria nel ciclo ha dato luogo a riporto e non è stato complementato.

Le cifre provenienti da Memoria possono essere complementate a 10 o a 9 o rimanere invariate in base allo stato dei FF Db, Da, dn.

Il FF Da si ripone all'uscita da Memoria del primo carattere ≠.O.

Il FF Db è riposto all'inizio di una fase β e può essere successivamente disposto o riposto dai seguenti segnali:

βw p_0 M_0 segno negativo dell'operando da Accumulatore

βw p_1 M_2 segno negativo del moltiplicatore in registro T

βw p_2 M_0 segno negativo del moltiplicando da Memoria.

Se il numero dei segnali è dispari il FF Db rimane disposto; se il numero dei segnali è pari rimane riposto.

Ognuno di questi segnali compare prima dell'uscita del primo carattere da Memoria.

Se il FF Db è riposto si ottiene per le cifre del moltiplicando quanto segue:

complementazione a 10 se disposti i FF Da, dn

complementazione a 9 se riposto il FF Da e disposto il FF dn

cifra invariata se riposto il FF dn.

Se il FF Db è disposto si ottiene invece:

complementazione a 10 se disposto il FF Da e riposto il FF dn

complementazione a 9 se riposti i FF Da, dn

cifra invariata se disposto il FF dn.

Sequenza delle operazioni a cominciare da p.d.c. p_8 della fase α

Durante la fase α si svolgono operazioni analoghe a quelle della maggior parte delle istruzioni, con trasferimento del carattere di funzione nei FF F, dell'indirizzo iniziale modificato nel registro O, e della lunghezza dell'operando di Memoria nei FF Lu, Ld.

Fase α

P_7 M_4 si azzera il registro JR interrogandolo
P_8 M_8 si ripone il FF Kz
M_9 si dispone il FF Po
M_9 si dispone il FF ρ (con $\alpha.P_8.\Delta AX$)
M_9 si ripone il FF α
M_9 si azzera il contatore Cp.

7.2. **Fase ρ**

p.d.c. P_0 M_0 si azzera l'indirizzatore V
M_0 l'indirizzatore W non conta — 1 e manca il pilotaggio di Memoria
M_4 si trasferisce all'indirizzatore V l'indirizzo 00 del registro JR (azzerato in $\alpha.P_7.M_4$). Si ha AJR = $\rho.P_0$; CJV = $\rho.P_0$
M_5 si dispone il FF Kz
M_9 si ripone il FF P_0
M_9 il contatore Cp conta + 1

p.d.c. P_1 M_0 l'indirizzatore V non conta (il FF λ è riposto da $\alpha.P_1.M_9$)
M_0 è presente il segnale PTI di consenso al pilotaggio dei registri TI; (è presente XAF = $\rho.P_1.\overline{\varnothing r}$)
M_3 si ripongono i FF Rm
M_4 si legge il carattere contenuto nel registro T all'indirizzo 00; viene rigenerato al successivo M_7 (CAR = ρ)
M_7 si trasferisce dall'indirizzatore V l'indirizzo 00 nel registro JR; (si ha AJR = $\rho.P_1$; CVJ = $\rho.P_1$)
M_9 il carattere letto nel registro T, attraverso il Ca, viene trasferito nei FF Rm. Il FF dn è riposto da $\alpha.P_2.M_2$; se il carattere uscito dal T è $\geqslant$ 5 (presenza del dAu) viene complementato a 10; si ha infatti: DA = $\rho.\overline{P_0}$.dAu; + 1 = $\overline{dn}.\rho.\overline{P_0}$.dAu

p.d.c. P_2 M_0 si azzera l'indirizzatore V, manca il pilotaggio di T

M_3 se il carattere uscito dal T in P_1 è $\geqslant 5$ si dispone il FF dn con $dAu.\rho.P_2$

M_4 si trasferisce all'indirizzatore V il contenuto del registro P; $AJP = \rho.P_2.\overline{\lambda}$; $CJV = \rho.P_2$

M_7 si riscrive nel registro P l'indirizzo di V; $CVJ = \rho.P_2.\overline{\lambda}$

M_8 si ripone il FF Kz

M_9 si ripone il FF ρ; si dispone il FF βw; si dispone il FF P_0; si azzera il contatore Cp

7.3. Fase β

p.d.c. P_0 M_0 si azzera l'indirizzatore W

M_0 manca il pilotaggio in Memoria ed in Accumulatore

M_4 si trasferisce il contenuto del registro 0 all'indirizzatore W

M_5 si dispone il FF Kz

M_7 si trasferisce il contenuto dell'indirizzatore W nel registro 0

M_9 si ripone il FF P_0; il contatore Cp conta + 1

p.d.c. P_1 M_0 l'indirizzatore W conta — 1

M_0 sono presenti i segnali di consenso al pilotaggio in Memoria e in Accumulatore

M_3 si ripongono i FF Up (Ud), ed i FF Au, si azzera il registro dei riporti Rr

M_4 si interroga il registro Q onde azzerarlo $AJQ = \beta w.P_1.\Delta A.\Delta X$

M_4 si legge in Memoria (il carattere letto viene rigenerato al successivo mastro M_7)

M_4 il carattere letto viene staticizzato nei FF Up (Ud)

M_4 si legge in Accumulatore all'indirizzo indicato dall'indirizzatore V. (Il carattere letto non verrà rigenerato al mastro M_7)

M_4 il carattere letto in Accumulatore viene staticizzato nei FF Au

M_4 i due caratteri letti in Memoria ed in Accumulatore entrano nell'unità aritmetica. Il carattere da Accumulatore non viene complementato, il carattere da

Memoria viene complementato nel modo precedentemente detto. Nell'unità aritmetica si esegue il prodotto tra la cifra contenuta in Rm e quella immessa da Memoria. La cifra unità del prodotto ottenuto viene immessa nel sommatore S_1 e ad essa viene sommata la cifra proveniente da Accumulatore. L'eventuale riporto non viene conteggiato in questo periodo di cifra ma nel successivo assieme al contenuto dal contatore riporti

P_1 M_7 la cifra decine del prodotto ottenuto passa nel contatore dei riporti Kr

M_7 si scrive in Accumulatore il risultato della prima operazione

M_7 si trasferisce il contenuto dell'indirizzatore V nel registro Q (azzerato al mastro M_4, p_1)
$CVJ = \beta w.P_1.\Delta A.\Delta X.$

In tal modo Q contiene, al pari di P, l'indirizzo iniziale dell'Accumulatore

M_8 il contatore dei riporti Kr conta il riporto ottenuto all'M_4

M_9 si dispone il FF λ.

Nota al p.d.c. P_1

Se da Memoria si è avuto un segno si dispone il FF Sm e, in modo analogo alla + MA, si rigenera il contenuto dell'Accumulatore; in P_2 inoltre non si avrà conta nell'indirizzatore V.

p.d.c. P_2 M_0 l'indirizzatore W conta — 1

M_0 l'indirizzatore V conta + 1 con $\Delta A.\beta w.\overline{\Delta E}.\overline{Sm}$

M_3 si ripongono i FF Up (Ud) ed i FF Accumulatore

M_3 si azzera il registro dei riporti Rr

M_4 si legge in Memoria (il carattere letto viene rigenerato al successivo mastro M_7)

M_4 il carettere letto viene staticizzato nei FF Up (Ud)

M_4 si legge in Accumulatore (il carattere letto non viene rigenerato al successivo mastro M_7)

M_4 il carattere letto viene staticizzato nei FF Au

M_4 i due caratteri letti in Memoria e in Accumulatore vengono introdotti nella U.A.D. passando attraverso

i complementatori Co e Ca. L'eventuale complementazione della cifra uscente da Memoria avviene nel modo precedentemente detto. L'U.A.D. opera sulle cifre provenienti dai FF Up (Ud), Au, Rm come nel precedente p.d.c.

P_2 M_5 il contatore dei riporti Kr trasferisce il suo contenuto ai FF Rr.

Il precedente riporto può sommarsi in S_2 al risultato del sommatore

M_5 si azzera il contatore dei riporti Kr

M_7 la cifra decine del risultato passa nel contatore di riporti Kr

M_7 si scrive in Accumulatore il risultato della 2ª operazione

M_8 il contatore Kr conteggia il riporto ottenuto all'M_4 da S_1, (l'eventuale riporto ottenuto in S_2 viene conteggiato all'M_1 del p.d.c. seguente).

Le operazioni proseguono nei successivi p.d.c. finchè non arrivano i segnali ∅c e ∅a indicanti la fine dei due numeri in Memoria ed in Accumulatore.

La fine del numero in Memoria può avvenire su carattere chiave colle stesse modalità specificate per la MA.
Facendo una analisi di tutti i casi possibili, analoga a quella fatta per la + MA, ci si può rendere conto che sono molto rari, e difficilmente decodificabili, i casi in cui si può finire senza l'estensione di un periodo di cifra; si è quindi preferito avere sempre l'astensione di un periodo di cifra, tranne due casi particolari che esamineremo in seguito.

Coi segnali ∅a. ∅cc. $\overline{\text{Ar}}$ si dispone quindi, all'M_4 del p.d.c. successivo, o periodo di cifra di estensione, il FF ∅p; con tale segnale all'M_7 si dispone il FF φp. Durante il p.d.c. di estensione non si ha pilotaggio di Memoria (per la presenza di ∅d), si ha pilotaggio di Accumulatore ma assenza del relativo strobe (per la presenza di ∅a).

Da Memoria si avrà quindi uno zero che dopo il Co rimarrà tale (se è $\overline{\text{IDM}}$) o diventerà 9 (IDM); da Accumulatore si avrà pure uno zero che dopo il Ca rimane tale (se è $\overline{\text{Cn}}$) o diventerà 9 (se è Cn).

Come si era accennato vi sono due casi in cui non si ha tale estensione; e precisamente quando, avendosi già estensione di Memoria (presenza di ∅d), nel p.d.c. in cui compare il gA:

a) esce uno zero (I0 φ) e vi è il segnale $\overline{Cn}$

b) esce un nove (I9 φ) e vi è il segnale Cn

E' ovvio che in tali due casi non è necessaria l'estensione dato che tale periodo di cifra corrisponde completamente alla situazione che si avrebbe con l'estensione.

Viene evitata l'estensione almeno in questi due casi altrimenti ogni fase β della moltiplicazione allungherebbe, anche senza necessità, di una cifra il contenuto dell'Accumulatore; ciò, dopo poche moltiplicazioni, potrebbe portare a un supero della massima capacità dell'Accumulatore.

Disponendosi all'M_7 il FF φp si susseguono le seguenti operazioni:

M_7 si ripone il FF ∅cc
M_8 si ripone il FF Kz con ΔAX.φp
M_9 si ripone il FF βw con ΔAX.φp.$\overline{\emptyset r}$
M_9 si dispone il FF ρ con ΔAX.φp.$\overline{\emptyset r}$
M_9 si azzera il contatore Cp e si dispone il FF P_0.

Ha inizio una seconda fase ρ.

7.4. 2ª Fase ρ

p.d.c. P_0 M_0 si azzera l'indirizzatore V
M_4 si trasferisce all'indirizzatore V l'indirizzo 00 del registro R
M_5 si dispone il FF Kz
M_9 si ripone il FF Po
M_9 il contatore Cp conta + 1

p.d.c. P_1 M_0 l'indirizzatore V conta + 1 con $\rho.P_1.\overline{Ar}.M_0.\lambda$
M_0 è presente il segnale P T_I
M_3 si ripongono i FF Rm
M_4 si trasferisce nei FF Rm il carattere contenuto nel registro T all'indirizzo 01
M_7 si trasferisce dall'indirizzatore V al registro R l'indirizzo 01

p.d.c. P_2 M_0 si azzera l'indirizzatore V

M_0 manca il segnale PT_I

M_4 si trasferisce all'indirizzatore V il contenuto del registro Q, cioè la $DA . AJQ = \rho . P_2 . \lambda$

M_8 si ripone il FF Kz

M_9 si ripone il FF ρ

M_9 si dispone il FF βw

M_9 si dispone il FF Po

M_9 si azzera il contatore Cp.

Poniamo in evidenza le differenze di questa seconda fase ρ rispetto alla prima, dovute alla presenza del segnale λ:

a) L'indirizzatore V conta + 1 in $P_1.M_0$. Ciò per spostarsi sulla cifra successiva del moltiplicatore contenuto nel T_0

b) Si seleziona il registro Q, al posto del registro P, in P_2; ciò per avere l'indirizzo di inizio dell'Accumulatore spostato ogni volta di una cifra a sinistra: dato però che nel Q, per questa seconda fase ρ, si ha proprio l'indirizzo della DA, si dovrà avere in $\beta w.P_1.M_0$ una conta nel V.

Nella fase β del secondo ciclo si svolgono le stesse operazioni indicate per la prima fase β; l'unica differenza, come già accennato in b), è che si ha una conta in più nell'indirizzatore V in $\Delta X.\Delta A.\beta w.\overline{\eta}.\overline{Ar}.P_1.M_0.\lambda$.

Quando si opera l'ultimo carattere del moltiplicatore si dispone il FF $\emptyset$r con ρ.gAu.PTS. Se l'ultimo carattere del moltiplicatore è minore di 5 non viene complementato, il FF dn se era disposto viene comunque riposto in $\rho.P_2.M_2$, e al termine della successiva fase β il segnale φp non dispone più il FF ρ, ma si dispone il FF φ, che pone fine alle operazioni dell'istruzione + X, con $\emptyset Am.\emptyset r.\overline{dn}.(\overline{I_X} + c\overline{Lu})$.

Se invece l'ultimo carattere del moltiplicatore è stato complementato, con la fase ρ si dispone il FF dn. Al mastro M_9 dell'ultimo p.d.c. della fase β si dispone ancora il FF ρ con φp.dn. Si opera un'altro ciclo durante la cui fase ρ non si estrae dal registro T_0 alcuna cifra per mancanza del pilotaggio (è presente $\emptyset$r), si trasferisce 1 nei FF Rm per la presenza del comando + 1 al Ca, e si ripone il FF dn.

Al termine della fase β si può infine disporre il FF φ.

Ovviamente, potendosi avere la fine su carattere chiave, nei casi Q, Z (cioè con fase η) si segue una procedura modificata; precisamente si ha:

a) viene impedito il disporsi del FF φp (solo alla fine dell'ultimo ciclo, cioè con $\emptyset r.\overline{dn}.cLu.\Delta AX$)

b) viene impedito il disporsi del FF φ e si dispone il FF η (analogamente alla MA).

7.5. **Esempi numerici**

1) Numero in Memoria 7687
   Numero in T 87
   Numero in A 36

1° ciclo Rm = 3 ; da Acc. 36 ; da Memoria 92313
(si è tenuto conto dell'estensione di una cifra)

3 × 3 = 9	9 + 6 = 5	$IRd_1 = 1$	
3 × 1 = 3	3 + 3 + 1 = 7	$IRd_1 = 0$;	$IRd_2 = 0$
3 × 3 = 9	9 + 0 = 9	$IRd_1 = 0$;	$IRd_2 = 0$
3 × 2 = 6	6 + 0 = 6	$IRd_1 = 0$;	$IRd_2 = 0$
3 × 9 = 27	7 + 0 = 7	$IRd_1 = 0$;	$IRd_2 = 0$

Si hanno IRd = 0; φ∅ = φp; XAY = IDM; e tutte le altre condizioni per disporre il FF Cn.

2° ciclo in M_2, po, ρ si dispone Cn

Rm = 1 ; da Acc. 97697 ; da Memoria 92313

1 × 3 = 3	3 + 7 = 0	$IRd_1 = 1$	
1 × 1 = 1	1 + 9 + 1 = 1	$IRd_1 = 1$;	$IRd_2 = 0$
1 × 3 = 3	3 + 6 + 1 = 0	$IRd_1 = 0$;	$IRd_2 = 1$
1 × 2 = 2	2 + 7 + 1 = 0	$IRd_1 = 0$;	$IRd_2 = 1$
1 × 9 = 9	9 + 9 + 1 = 9	$IRd_1 = 1$	$IRd_2 = 0$

Il FF Cn non può comunque essere riposto essendo assente XAλ = $\overline{IDM}$.

3° ciclo Si ha un terzo ciclo essendovi ∅r . dn

Rm = 1 ; da Acc. 99001 ; da Memoria 07687

1 × 7 = 7	7 + 1 = 8	$IRd_1 = 0$	
1 × 8 = 8	8 + 0 = 8	$IRd_1 = 0$;	$IRd_2 = 0$

$1 \times 6 = 6$	$6 + 0 = 6$	$IRd_1 = 0$;	$IRd_2 = 0$
$1 \times 7 = 7$	$7 + 9 = 6$	$IRd_1 = 1$;	$IRd_2 = 0$
$1 \times 0 = 0$	$0 + 9 + 1 = 0$	$IRd_1 = 0$	$IRd_2 = 1$

Risultato in accumulatore 0668805 in forma normale.

Essendovi: $XA\lambda = \overline{IDM}$; $IRd = IRd_2$; e gli altri segnali necessari, all'M_2 del successivo p.d.c. $(\alpha . P_0)$ si ripone il FF Cn.

2) Numero in Memoria 4715
Numero in T — 16
Numero in A — 12650

Essendovi due segni negativi si ha $\overline{Db}$.

1° ciclo Rm = 4 ;	da Acc. 012650 ;	da Memoria 995285	
$4 \times 5 = 20$	$0 + 0 = 0$	$IRd = 0$	$Kr = 2$
$4 \times 8 = 32$	$2 + 5 + 2 = 9$	$IRd = 0$	$Kr = 3$
$4 \times 2 = 8$	$8 + 6 + 3 = 7$	$IRd = 1$	$Kr = 1$
$4 \times 5 = 20$	$0 + 2 + 1 = 3$	$IRd = 0$	$Kr = 2$
$4 \times 9 = 36$	$6 + 1 + 2 = 9$	$IRd = 0$	$Kr = 3$
$4 \times 9 = 36$	$6 + 0 + 3 = 9$	$IRd = 0$	$Kr = 3$

Esistono le condizioni necessarie per disporre il FF Cn.

2° ciclo Rm = 2 ;	da Acc. 99379 ;	da Memoria 04715	
$2 \times 5 = 10$	$0 + 9 = 9$	$IRd = 0$	$Kr = 1$
$2 \times 1 = 2$	$2 + 7 + 1 = 0$	$IRd = 1$	$Kr = 1$
$2 \times 7 = 14$	$4 + 3 + 1 = 8$	$IRd = 0$	$Kr = 1$
$2 \times 4 = 8$	$8 + 9 + 1 = 8$	$IRd = 1$	$Kr = 1$
$2 \times 0 = 0$	$0 + 9 + 1 = 0$	$IRd = 1$	$Kr = 1$

Esiste la condizione per finire anche senza estensione dell'Accumulatore (si ha $I9\varphi . Cn$).

Esistono le condizioni per riporre il FF Cn.

Risultato in Accumulatore 083090; numero in forma normale, segnato negativo.

7.6. **Istruzione — X**

Decodificato ΔAXM; ΔK

E' perfettamente analoga alla + X; l'unica differenza è costituita dal fatto che il FF Db, connesso a contatore, in βw; P_0; M_6 (col segnale $\overline{\eta} . \Delta M . P_0 . M_6$) riceve un impulso ulteriore di conta, che tiene conto del segno negativo dell'operazione.

8. ALTRE ISTRUZIONI ARITMETICHE

8.1. Istruzioni + MT e — MT

Decodificati istruzione + MT : $\Delta T \Sigma P$; ΔK
Decodificati istruzione — MT : $\Delta T \Sigma M$; ΔK.

La fase α di tali istruzioni, essendovi il decodificato ΔT, è perfettamente identica a quella dell'istruzione MT. Si ottiene cioè il trasferimento dell'indirizzo iniziale dell'operando da Memoria nel registro 0, l'indirizzo del registro T su cui si deve operare nel V, la lunghezza nel FF Lu.

Fase β. Gli organi logici che intervengono nell'esecuzione sono: l'Unità Aritmetica, il FF Db, il FF Ot, oltre agli organi relativi alla estrazione dei dati da Memoria e da registri T.

Il FF Ot disponendosi indica che si è avuto « overflow » nel registro T su cui si è operato. Si avrà tale indicazione:

a) se il risultato, per effetto di un riporto, è più lungo del più lungo dei due operandi (nel caso della + MT); il riporto viene scritto nel registro T

b) se il minuendo è minore del sottraendo e quindi si ha il risultato in complemento (nel caso della — MT)

p.d.c. P_0 M_0 si azzera l'indirizzatore W

M_4 si trasferisce all'indirizzatore W il contenuto del registro 0

M_5 si dispone il FF Kz

M_6 si dispone il FF Db (solo nella — MT con $\Delta M.P_0.\overline{\eta}.M_6$)

M_7 l'indirizzo letto in M_4 viene riscritto nello stesso registro 0

M_8 si ripone il FF Ot con $\Delta T.\Delta\Sigma.\overline{\Delta\Sigma B}.\beta r.P_0.\overline{\eta r}$

M_9 il contatore Cp conta + 1

p.d.c. p_1 M_0 l'indirizzatore W conta — 1

è presente il segnale pMp (pMd) di consenso al pilotaggio della Memoria e il segnale XAF di consenso al pilotaggio dei registri ($XAF = \Delta T.\beta r.\overline{\Delta B}.\overline{\Delta AER}$)

M_4 si estrae da Memoria il primo carattere; non è ammesso che sia un segno. Esso viene staticizzato nei FF Up (Ud) e verrà rigenerato in Memoria all'M_7 successivo. Si estrae dal registro T la prima cifra dell'operando e la si staticizza nei FF Au.

Il carattere da Memoria, complementato o meno, e il carattere da registro T, inalterato, vengono operati nell'U.A.D.

M_7 si scrive nel registro T il risultato delle prime due cifre operate

M_9 il contatore Cp conta + 1.

Nei periodi di cifra successivi si opera in modo del tutto uguale a quello descritto per il p.d.c. P_1, avendosi ad ogni M_0 una conta —1 nell'indirizzatore W e una conta +1 nell'indirizzatore V.

La fine dell'istruzione è data dalla presenza di entrambi i segnali ∅c, ∅a.

Se il numero proveniente da Memoria è più lungo di quello proveniente dal registro T con la presenza del bit ga si dispone il FF ∅a, e si inibisce lo strobe per i p.d.c. successivi, si avrà quindi uscita zero dal registro T.

Se invece il numero proveniente dal registro T è più lungo di quello proveniente dalla Memoria, dopo che è stato letto in Memoria l'ultimo carattere si ha il segnale ∅c che dispone all'M_9 il FF ∅d. Tale segnale ∅d inibisce il pilotaggio di Memoria; dalla Memoria escono pertanto zeri.

All'apparire di entrambi i segnali ∅c, ∅a si decide se si può finire immediatamente, o solo dopo un ulteriore p.d.c., in base al seguente specchietto, analogo a quello della + MA:

Db		$\overline{Db}$	
fine immediata	fine immediata, disporre Ot	prolungamento 1 p.d.c., disporre Ot	fine immediata
IRφ	$\overline{IR\varphi}$	IRφ	$\overline{IR\varphi}$

Si può quindi avere la fine dell'operazione nei casi: $Db . IR\varphi + \overline{Db} . \overline{IR\varphi}$.

Il FF Ot è disposto nei due casi $\overline{Db}.IRd + Db.\overline{IRd}$ all'M_9 col segnale $\Delta T . \overline{\Delta\Sigma B} . \varnothing a . (\varphi + \varnothing c)$.

Valgono le regole di fine su carattere chiave, ovviamente solo per i casi Z, Y.

8.2. **Istruzione + MM e — MM**

Decodificati + MM : $\Delta\Sigma P$; ΔK

Decodificati — MM : $\Delta\Sigma M$; ΔK.

Si ricorda che tali operazioni possono avvenire solo con quantità non segnate.

La fase α di tali istruzioni è perfettamente identica a quella della istruzione MEM.

Si ha cioè azzeramento del registro di lunghezza solo se vi è # in Ld, dato che la lunghezza può essere specificata già nella PUM che le deve precedere.

Fase β. Gli organi logici che intervengono nell'esecuzione sono: l'U.A., il FF Db, il FF Om, oltre agli organi relativi all'estrazione dei dati da Memoria su entrambi i canali.

Il FF Om disponendosi indica che si è avuto « overflow » nella Memoria su cui si operato. Si avrà tale indicazione:

a) se il risultato, per effetto di un riporto, è più lungo di LL. Il riporto non viene scritto nella posizione adiacente di Memoria (nel caso della + MM);

b) se il minuendo è minore del sottraendo e quindi si ha il risultato in complemento.

Valgono le regole relative alla fine su carattere chiave: quest'ultimo deve essere contenuto nella zona di Memoria con indirizzo specificato nella — MM, in cui si forma il risultato (BMI).

p.d.c. P_0 M_0 si azzera l'indirizzatore W

M_4 si trasferisce all'indirizzatore W il contenuto del registro 0

M_5 si dispone il FF Kz

M_6 si dispone il FF Db (solo nella —MM con $\Delta M.P_0.\overline{\eta}.M_6$)

M_7 si riscrive nel registro 0 l'indirizzo letto all'M_4
M_8 si ripone il FF Om con $\Delta S\Sigma P$ (o $\Delta S\Sigma M$)$.\beta w.P_0.\overline{\eta}$.
M_8 si dispone il FF Ce con $\Delta MM.\beta w.P_0$
M_9 il contatore Cp conta + 1

p.d.c. P_1 M_0 l'indirizzatore W conta — 1; l'indirizzatore Y conta — 1; sono presenti entrambi i pilotaggi di Memoria; per il canale esterno coi segnali $\beta y.Ce.\overline{XW\lambda}$; per il canale interno coi segnali $XMX.\overline{XMY}.\overline{\varnothing d}.\overline{\eta}$

M_4 si estraggono da Memoria, canale esterno ed interno, i primi due caratteri; quello relativo al canale esterno viene rigenerato in Memoria al successivo M_7; esso viene introdotto all'ingresso del Ca. Nel caso che entrambi gli indirizzi siano pari o dispari, mediante i FF Ar e Am si opera in modo analogo alla MEM.

Il carattere uscito sul canale interno, complementato o meno nel Co, viene operato nell'U.A. insieme al carattere uscito sul canale esterno (indirizzo specificato nella PUM)

M_7 si scrive in Memoria, canale interno, il risultato ottenuto in U. A. tramite il segnale $XMJ = \Delta MM.\overline{\Delta SEB}.\beta r.\overline{XM\alpha}$.

Nei p.d.c. successivi si opera in modo del tutto uguale.

La fine dell'istruzione è data dal comparire del segnale $\varnothing c$; si ottiene il consenso al set del FF φ: $XFH = \Delta MM.\varnothing c$.

Se vi è fine su carattere chiave, il carattere stesso, trovato sul canale interno viene rigenerato (per la presenza del segnale $XM\alpha$ che impedisce la scrittura del risultato ottenuto in U.A. ed abilita la rigenerazione del canale interno).

Si deve disporre il FF Om nei casi in cui:

a) si è avuto riporto nell'ultimo p.d.c. operativo se vi è $\overline{Db}$

b) non si è avuto riporto nell'ultimo p.d.c. operativo se vi è Db.

Il riporto nell'ultimo p.d.c. operativo è proprio quello dell'ultimo p.d.c. se non si ha fine su carattere chiave, altrimenti è quello del p.d.c. precedente; si deve notare che il riporto eventuale del p.d.c. precedente è staticizzato nel registro dei riporti Rr.

Si ha quindi il segnale: $XA\Omega = IR\varphi.\overline{XM\alpha} + aRr.XM\alpha$; tale segnale indica che si è avuto riporto nell'ultimo p.d.c. operativo.
Si dispone quindi il FF Om nei casi:

$$(\overline{Db}.XA\Omega + Db.\overline{XA\Omega})XA\varphi$$

dove è $XA\varphi = \not{c}c + \varphi$.

8.3. **Istruzione + AM**

Decodificati $\Delta A \Sigma B$; ΔK.

La fase α, essendovi il decodificato ΔA, è identica a quella dell'istruzione MA.

Fase β. Gli organi logici che intervengono nell'esecuzione sono: l'U.A., il FF Db, il FF Om, oltre agli organi relativi all'estrazione dei dati da Memoria e da accumulatore.
Per quanto riguarda i segni si deve notare:

a) se la Memoria è segnata il segno di memoria non è preso in considerazione (cioè non influisce sul FF Db per l'assenza di $\overline{\Delta\Sigma B}$) e rimane invariato

b) il segno in accumulatore influisce sul FF Db.

L'operazione si arresta sempre alla lunghezza LL, anche se il contenuto dell'accumulatore è più lungo.

Se il contenuto dell'accumulatore è più corto viene esteso alla lunghezza LL con zeri se è $\overline{Cn}$, o con 9 se vi è Cn.

Il FF Om viene disposto:

a) se nell'ultimo p.d.c. operativo vi è riporto ($XA\Omega$) ed è $\overline{Db}$
b) se nell'ultimo p.d.c. operativo non vi è riporto ($\overline{XA\Omega}$) ed è Db

p.d.c. P_0 M_0 si azzera l'indirizzatore W

M_0 si dispone il FF Db (se l'accumulatore è segnato negativo)

M_4 si trasferisce all'indirizzatore W il contenuto del registro 0

M_5 si dispone il FF Kz

M_7 si riscrive nel registro 0 l'indirizzo letto all'M_4

M_8 si ripone il FF Om con $\Delta A\Sigma B.P_0.\beta_r.\overline{\eta_r}$.

M_9 il contatore Cp conta + 1

p.d.c. P_1 M_0 l'indirizzatore W conta — 1

è presente il segnale pMp (pMd) di consenso al pilotaggio della Memoria, e il segnale PA di consenso al pilotaggio dell'accumulatore ($PA = \Delta B . \Delta A . \overline{\Delta E} . \overline{\varnothing a}$)

M_4 si estrae da Memoria il primo carattere e viene staticizzato nei FF Up (Ud).

Si estrae da accumulatore il primo carattere e viene staticizzato nei FF Au; verrà rigenerato al successivo M_7 ($CAR = \beta r . \Delta\Sigma . \Delta B$)

Il carattere da Memoria, complementato o meno, e il carattere da accumulatore vengono operati nell'U.A.

M_7 l'uscita dell'U.A. viene scritta in Memoria mediante il segnale $XMJ = \Delta\Sigma . \Delta B . \beta r . \overline{XM\alpha} . \overline{XML}$ dove è $XML = Sm.P_1R$

M_9 il contatore Cp conta + 1.

Nota p.d.c. P_1

Se da Memoria esce un segno manca il segnale XMJ per la presenza di XML; il segno viene riscritto in Memoria mediante il segnale $XMG = \Delta\Sigma B . \beta r . Sm . P_1 R$.

I successivi periodi di cifra sono identici al primo; vi è inoltre ad ogni M_0 la conta + 1 nell'indirizzatore V; tale conta non si ha in P_2 se in P_1 è uscito da Memoria un segno.

Se compare il bit gA prima che si sia raggiunta la lunghezza LL, nei p.d.c. successivi manca il pilotaggio di accumulatore (per la presenza di $\varnothing$a).

Si avrà complementazione dello zero bit presente nei FF Au se vi è Cn.

$DA = \Delta K . \overline{gAu} . \varnothing a . XAQ$; dove è $XAQ = \Delta A . \beta r . Cn . \overline{\eta r}$.

La presenza del segnale $\varnothing$c dà immediatamente il consenso al disporsi del FF φ; è $XFH = \varnothing c . \Delta B . \Delta\Sigma$.

Si deve notare che nei casi di fine su carattere chiave si ha rigenerazione dello stesso in Memoria; ciò mediante il segnale $XM\alpha$ nel modo già visto per la + MM.

Per quanto riguarda l'eventuale disporsi del FF Om valgono le stesse considerazioni effettuate per la + MM.

Si deve notare qui che la presente istruzione quando l'accumulatore è segnato ed in complemento e vi è segno in Memoria può dare un risultato non matematicamente corretto.

Esempio numerico

In accumulatore	8 7 2	segnato negativo in complemento
In Memoria	1 2 2 —	

L'accumulatore contiene in realtà la quantità 1 2 8 + in vera grandezza; il risultato dovrebbe quindi essere 6 +. Si ottiene invece:

da accumulatore	8 7 2	
da Memoria	8 7 8 —	(per la presenza di Db; segno negativo in accumulatore)
risultato in Memoria	7 5 0 —	

8.4. **Istruzione + TM**

Decodificati ΔTΣB; ΔK.

La fase α, essendovi il decodificato ΔT, è identica a quella della MT.

La fase β è identica a quella dell'istruzione + AM, tenendo presente che le quantità non possono essere segnate. La fine su carattere chiave può aversi ovviamente solo per i casi Z, Y.

9. ISTRUZIONI LOGICHE

9.1. Istruzione + LD

Configurazione L Tb IIII Ta V. Decodificato ΔTXR.

Si esegue la somma logica diretta, bit per bit, tra il dato di Memoria, ed il dato contenuto nel registro Tb.

La lunghezza del dato nel Tb è individuata dal bit gT.

La fase α è identica a quella dell'istruzione MT.

Fase β: si pongono in evidenza solo le differenze rispetto alla MT. I caratteri estratti da Memoria e dal registro Tb vengono operati in una rete logica apposita. I risultati vengono scritti nel registro Tb.

La fine dell'istruzione avviene appena finisce il più corto dei due operandi; si ha infatti il consenso al disporsi del FF φ col segnale $XFH = \emptyset a.\Delta T.\Delta X.\beta w + \emptyset c.\Delta T.\Delta X$.

Per la presenza del segnale ΔTX l'uscita dell'U.A. è bloccata.

La scrittura nel registro Tb è condizionata dal segnale $XA\Theta = \Delta T.\Delta X.\beta r.\overline{XM\alpha}.\overline{\eta r}$; quindi nel caso di fine su carattere chiave (ovviamente solo Z, Y) nell'ultimo p.d.c. si scriverà uno zero (essendovi $XM\alpha$); ovviamente si scriverà l'or dei BMI e dei BAu, condizionato al decodificato ΔR.

Nell'ultimo p.d.c. si scrive anche il bit gT disponendo all'M_7 il FF Ga col segnale $\emptyset a.\Delta TX.\overline{\eta} + \emptyset c.\Delta TX.\overline{\eta}$.

9.2. Istruzione XLD

Configurazione L Tb IIII Ta! Decodificato ΔTXP.
Si esegue la moltiplicazione logica diretta, bit per bit, tra un dato contenuto in Memoria ed un altro contenuto in un registro Tb.

Lo svolgimento dell'istruzione è del tutto analogo a quello della + LD; si scrive nel registro Tb l'and dei BMI e dei BAu, condizionato da ΔP.

9.3. **Istruzione XLN**

Configurazione L Tb IIII Ta &   Decodificato ΔTXM

Si esegue la moltiplicazione logica negata, bit per bit, tra un dato contenuto in Memoria ed un altro contenuto in un registro Tb.

Lo svolgimento dell'istruzione è del tutto analogo a quello della + LD; si scrive nel registro Tb l'and dei $\overline{\text{BMI}}$ e dei $\overline{\text{BAu}}$, condizionato da ΔM.

10. ISTRUZIONE IT

10.1. Considerazioni generali

Decodificati ΔSXB; ΔI; ΔK   Configurazione #CCCCC Ta 9

Con la istruzione IT si trasferisce nel registro Ta la costante CCCCC; il bit gT viene posto sulla quinta posizione del registro.

Se nelle posizioni P_7 e P_8 è registrato il carattere ÷ il bit gT è posto sulla quarta posizione e la quinta posizione del registro viene azzerata.

10.2. Fase α

È identica a tutte le fasi α per i p.d.c. P_0, P_1, P_2; nel p.d.c. P_2 si ha inoltre:

p_2 M_9 si ripone il FF α con $P_2.\Delta I$
M_9 si dispone il FF βw con $P_2.\overline{\Delta AX}.\Delta I$

Si deve notare che non si ripone il FF Kz; la fase β perciò inizia dal p.d.c. P_3. Inoltre il FF α_{37} è disposto in $\alpha.P_2.M_9$ e rimane disposto fino al p.d.c. P_7 della fase β.

10.3. Fase β

p.d.c. p_3 M_0 l'indirizzatore V non conta (conta con $\alpha_{37}.P_3$)
l'indirizzatore W conta — 1
è presente il segnale pMp (pMd) di consenso al pilotaggio della Memoria
è presente il segnale di pilotaggio dei registri T; (con $\alpha_{37}.\overline{\phi a}$)

M_4 si legge in Memoria e il carattere letto è staticizzato nei FF Up (Ud), si legge nel registro T, il carattere letto non viene estratto essendo inibito lo strobe da $\Delta SXB.\beta w.\overline{P_8}$

M_7 si trasferisce nel registro T il carattere letto in Memoria; esso attraversa inalterato l'U.A.;
CUA = ΔS.ΔX.βr
si rigenera in Memoria il carattere letto all'M_4

M_9 il contatore Cp conta + 1

p.d.c. p_4 p_5 p_6 si opera come al p.d.c. P_3 con conta indietro dell'indirizzatore W e conta in avanti dell'indirizzatore V

p.d.c. p_7 si opera come ai p.d.c. precedenti se da Memoria non esce il carattere ÷
il carattere ÷ da Memoria blocca l'uscita del Co per cui si scrive zero nel T. (manca CMU per causa del segnale ISM.ΔI, dove ISM decodifica l'uscita di Memoria ÷ o #.)

p.d.c. p_8 M_0 l'indirizzatore W conta — 1
l'indirizzatore V non conta; esso è arrivato alla 5ª posizione del registro, quindi in Vu vi sarà 4 o 9; se nel p.d.c. P_7 è uscito ÷ da Memoria si porta l'indirizzatore V sulla quarta posizione del registro, cioè rispettivamente all'indirizzo 3 o 8; ciò equivale a disporre i FF a e b del Vu, col segnale:

$\Delta I.ISM.dMI.XWW.M_0$ dove $XWW = P_8$; ISM.dMI decodifica il ÷

M_4 si legge in Memoria (# o ÷) e l'uscita è staticizzata; ne è inibito il passaggio attraverso il Co per la presenza del segnale ISM; si legge nel registro T e non essendo inibito in P_8 lo strobe, l'uscita viene staticizzata nei FF Au

M_7 si riscrive nel T (attraverso l'U.A.) il carattere uscito all'M_4, cioè quello che vi è stato scritto in P_7 (o in P_6 se in P_7 è uscito ÷); si aggiunge il bit gT disponendo il FF Ga con $\Delta I.P_8.M_7$

M_7 si dispone il FF φ con $\Delta I.P_8$; si ripone il FF Kz

M_9 si azzera il contatore Cp, si dispone il FF P_0 e il FF α.

11. UNITÀ ARITMETICA BIT e ED f

11.1. Considerazioni generali

L'unità aritmetica dei bit e ed f serve principalmente per la modifica degli indirizzi durante la fase α e per l'esecuzione di alcune particolari istruzioni tipo ΔI (+IT e —IT).

Ricordiamo che i bit e ed f ai fini dell'indirizzamento della Memoria hanno un « peso » in una aritmetica in base 4. Ad esempio i bit e ed f delle migliaia indicano le decine di migliaia; i bit e ed f delle centinaia indicano il « modulo » di Memoria.
La corrispondenza fra codice e « peso » è la seguente:

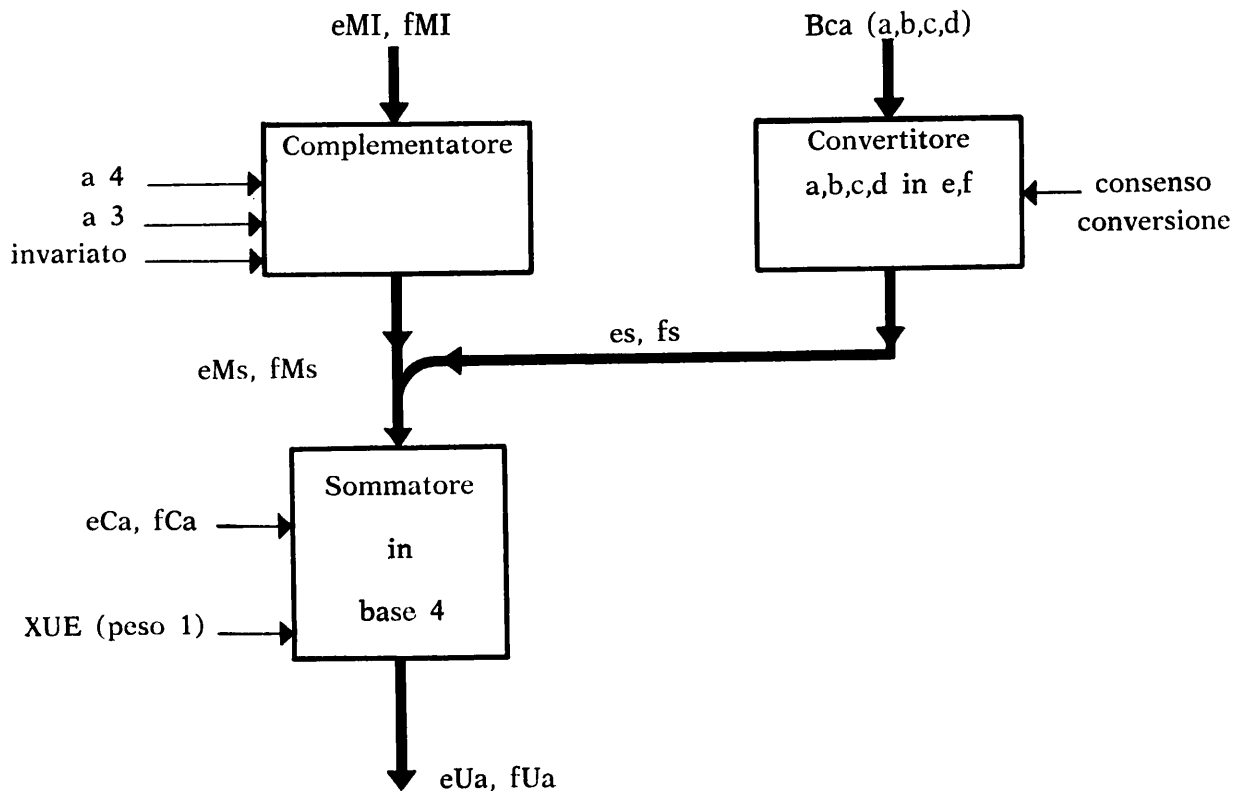
f	e	peso
0	0	0
0	1	1
1	1	2
1	0	3

Per operare aritmeticamente su tali « pesi » occorrerà quindi:

a) un complementatore di Memoria (analogo al Co) con complementazione a 4 (analoga di quella a 10) o a 3 (analoga di quella a 9)

b) un sommatore in base 4.

In alcune istruzioni sorge inoltre la necessità di trasformare una cifra espressa in forma normale (cioè con i bit a, b, c. d in base 10) in una cifra espressa coi bit e ed f in base 4.

Si ottiene così il seguente schema a blocchi.

11.2. **Complementatore**

I bit e ed f presenti all'ingresso dovranno essere trasformati in uscita, a seconda del comando presente, nel modo seguente:

Ingressi		a 4		a 3	
Peso	f, e	Peso	f, e	Peso	f, e
0	0 0	0	0 0	3	1 0
1	0 1	3	1 0	2	1 1
2	1 1	2	1 1	1	0 1
3	1 0	1	0 1	0	0 0

Si può notare che complementare a 4 corrisponde a scambiare i bit e ed f fra loro; complementare a 3 vuol dire lasciare invariato il bit e ed invertire il bit f.

11.3. **Convertitore**

Deve rispettare la seguente tabella:

Peso	Ingresso d c b a	Uscita f e
0	0 0 0 0	0 0
1	0 0 0 1	0 1
2	0 0 1 1	1 1
3	0 1 1 1	1 0
4	0 1 0 0	0 0

Si ottiene quindi:

$$e = a.\bar{c} \qquad ; \qquad f = b$$

È chiaro che tale conversione è corretta solo per i numeri fino a 4.

11.4. **Sommatore**

Gli ingressi di tale sommatore sono tre:

a) i bit e_{MS}, f_{MS} provenienti dal complementatore o dal convertitore con valore da 0 a 3

b) i bit provenienti dal Ca con valore da 0 a 3

c) il segnale XUE con valore 1 se presente, 0 se assente.

I bit eUa, fUa in uscita sono ottenuti secondo le seguenti tavole:

Con presenza del segnale XUE (peso 1)

da Memoria

da Accumulatore	0 1 2 3	0 1 2 3	0 1 2 3
0	1 2 3 0	1 1 0 0	0 1 1 0
1	2 3 0 1	1 0 0 1	1 1 0 0
2	3 0 1 2	0 0 1 1	1 0 0 1
3	0 1 2 3	0 1 1 0	0 0 1 1
	risultato	eUa	fUa

Con assenza del segnale XUE

da Memoria

da Accumulatore	0 1 2 3	0 1 2 3	0 1 2 3
0	0 1 2 3	0 1 1 0	0 0 1 1
1	1 2 3 0	1 1 0 0	0 1 1 0
2	2 3 0 1	1 0 0 1	1 1 0 0
3	3 0 1 2	0 0 1 1	1 0 0 1
	risultato	eUa	fUa

Si ha una prima decodificazione dei segnali:

$\mathrm{XUF} = \overline{\mathrm{XUE}}.\overline{\mathrm{eCa}}.\overline{\mathrm{fCa}}$ (zero da accumulatore, assenza di XUE)

$\mathrm{XUG} = \overline{\mathrm{XUE}}.\mathrm{eCa}.\overline{\mathrm{fCa}}$ (1 da accumulatore, assenza di XUE)

$\mathrm{XUH} = \overline{\mathrm{XUE}}.\mathrm{eCa}.\mathrm{fCa}$ (2 da accumulatore, assenza di XUE)

$\mathrm{XUI} = \overline{\mathrm{XUE}}.\overline{\mathrm{eCa}}.\mathrm{fCa}$ (3 da accumulatore, assenza di XUE)

$\mathrm{XUL} = \mathrm{XUE}.\overline{\mathrm{eCa}}.\overline{\mathrm{fCa}}$ (zero da accumulatore, presenza di XUE).

Si ottengono quindi le equazioni logiche:

$$\mathrm{eUa} = \mathrm{XUF.eMs} + \mathrm{XUG}.\overline{\mathrm{fMs}} + \mathrm{XUH}.\overline{\mathrm{eMs}} + \mathrm{XUI.fMs} + \mathrm{XUL}.\overline{\mathrm{fMs}} + \mathrm{XUE.eCa}.\overline{\mathrm{fCa}}.\overline{\mathrm{eMs}} + \mathrm{XUE.eCa.fCa.fMs} + \mathrm{XUE}.\overline{\mathrm{eCa}}.\mathrm{fCa.eMs}$$

$$\mathrm{fUa} = \mathrm{XUF.fMs} + \mathrm{XUG.eMs} + \mathrm{XUH}.\overline{\mathrm{fMs}} + \mathrm{XUI}.\overline{\mathrm{eMs}} + \mathrm{XUL.eMs} + \mathrm{XUE.eCa}.\overline{\mathrm{fCa}}.\mathrm{fMs} + \mathrm{XUE.eCa.fCa}.\overline{\mathrm{eMs}} + \mathrm{XUE}.\overline{\mathrm{eCa}}.\mathrm{fCa.fMs}$$

11.5. **Il comportamento nelle istruzioni non ΔI**

In tali istruzioni non agisce mai il convertitore.

Se è presente il segnale $\overline{\mathrm{IDM}}$ (cioè non si complementano i bit numerici) non si ha complementazione.

Se è presente o IDM o ID1M (complementazione a 9 o a 10) si complementa sempre a 4; per tali casi si ottiene perciò:

$\mathrm{eMS} = \mathrm{eMI}.\overline{\mathrm{IDM}} + \mathrm{fMI.XUQ}.\overline{\Delta\mathrm{I}}$
$\mathrm{fMS} = \mathrm{fMI}.\overline{\mathrm{IDM}} + \mathrm{eMI.XUQ}.\overline{\Delta\mathrm{I}}$
dove è $\mathrm{XUQ} = \mathrm{IDM} + \mathrm{ID1M}$.

Durante la fase α si tiene conto dell'eventuale riporto numerico nelle migliaia (cioè in P_6) aggiungendo un 1 alla parte letterale (cioè al valore dei bit e ed f), dato che tale parte rappresenta proprio le decine di migliaia; si ha quindi: $\mathrm{XUE} = \alpha.P_6\mathrm{R.IRd}$.

Nella IT, avendosi $\overline{\mathrm{IDM}}$ i bits e, f eventualmente provenienti da Memoria passano invariati nel complementatore e nel sommatore in base 4.

12. ISTRUZIONI + IT, — IT e ITT

12.1. Istruzione + IT (÷ CCCCC T +)

Decodificati ΔSXP; ΔI; ΔK

Con la istruzione + IT si esegue la somma dell'indirizzo contenuto nel registro T individuato al secondo p.d.c. della fase α con la costante CCCCC estratta da Memoria. Il risultato viene immesso nello stesso registro T.

Vengono presi in esame i seguenti casi:

a) la costante in Memoria è formata da 4 caratteri seguiti da due caratteri ÷. Nel registro T è contenuto un numero di 5 cifre.

Memoria ÷ ÷ CCCC T +

Registro T $\dot{C}$CCCC

In questo caso si ottiene il risultato nel registro T come indirizzo di 4 caratteri.

b) la costante in Memoria è formata da 4 caratteri seguiti da due caratteri ÷. Nel registro T è contenuto un numero di 4 caratteri o meno di 4.

In questo caso il contenuto di T risulta un indirizzo di 4 caratteri.

Memoria ÷ ÷ CCCC T +

Registro T $\dot{C}$CCC

c) la costante in Memoria è formata da 4 caratteri seguiti da due caratteri #. Nel registro T è contenuto un numero di 4 caratteri.

Memoria # # CCCC T +

Registro T $\dot{C}$CCC

In questo caso il contenuto di T risulta un numero di 4 caratteri, l'eventuale riporto fra i p.d.c. p_6 e p_7 non viene operato.

Il bit gt viene posto in 5ª posizione ed in essa viene scritto uno zero.

d) la costante in Memoria è un numero di 5 cifre seguite dal carattere #.

Nel registro T è contenuto un numero di lunghezza qualsiasi.

Memoria # CCCCC T +

Si ha l'indicazione di « overflow » se c'è riporto tra i p.d.c. p_7, p_8, detto riporto non viene operato. Il bit gt viene posto in 5ª posizione.

Esecuzione dell'istruzione + IT

La fase α dell'istruzione + IT termina come nell'istruzione IT al p.d.c. p_2. Le operazioni che si svolgono nella fase α sono uguali a quelle descritte nell'istruzione IT. Inoltre il FF Ot viene riposto con $\Delta I.\overline{\Delta B}.\alpha.P_2.M_8$.

La fase β dura sei p.d.c. Le operazioni relative alla fase β differiscono nei casi a, b, c, d, precedentemente considerati.

12.2. **Caso a** (÷ ÷ CCCC T +)

Fase β p.d.c. p_3, p_4, p_5

l'indirizzatore W conta — 1 ad ogni p.d.c.
l'indirizzatore V conta + 1 in p_4 e p_5, non conta in p_3
i caratteri estratti da Memoria e da registro T si sommano nell'U.A.D. Al mastro M_7 di ogni p.d.c. si scrive il risultato nel registro T.

In p_5 il carattere letto in Memoria, o nel registro, o in entrambi, può essere alfabetico; in questo caso i bits e ed f, passano invariati nel complementatore relativo (per la presenza di $\overline{IDM}$), e si sommano nel sommatore in base 4; il risultato viene scritto, insieme ai bits a, b, c, d, all'M_7 nel registro. Si ottiene così l'eventuale somma dei « moduli », in base 4.

p.d.c. p_6 l'indirizzatore W conta — 1. L'indirizzatore V conta + 1. Il carattere letto in Memoria (eventualmente anche nel registro) può essere alfabetico.

In questo caso i bits e ed f vengono operati come al p.d.c. p_5. I bits a, b, c, d vengono operati nor-

malmente nell'U.A.D. Il risultato viene scritto nel registro T.

Se l'operazione ha dato riporto nell'U.A.D., si dispone all'M_9 il FF λ con $\Delta I.p_6.IRd$.

p_7 l'indirizzatore W conta — 1. L'indirizzatore V conta + 1.

Il carattere ÷ letto in Memoria non attraversa il Co per assenza del segnale CMU (a causa di ISM.ΔI).

Il carattere letto nel registro T viene riscritto inalterato attraverso l'U.A.D. L'eventuale riporto relativo al p.d.c. precedente non viene conteggiato per la presenza del segnale ISM che azzera il registro dei riporti all'M_6.

Al mastro M_9 si effettua la conversione del numero uscito dal registro disponendo se necessario i rispettivi FF es, fs con le condizioni viste quando si è trattato il convertitore stesso, ed inoltre col consenso:

$\Delta I.\Delta P.P_7.ISM.dMI$ (cioè ÷ in P_7 di una + IT).

p_8 l'indirizzatore W conta — 1. L'indirizzatore V non conta e il suo contenuto viene ridotto di 1 in modo analogo a quanto visto nella IT.

Il carattere ÷ da Memoria non viene operato. Dal registro T viene estratto il carattere scrittovi all'M_7 del p.d.c. P_6. Esso viene modificato nei bits e ed f aggiungendovi nel sommatore in base 4 il peso dei FF e_s, f_S e l'eventuale riporto relativo al p.d.c. P_6, non operato al p.d.c. P_7, e staticizzato nel FF λ.

Si ha: $eMs = es.ISM.dMI.P_8R$
$fMs = fs.ISM.dMI.P_8R$
$XUE = \lambda .ISM.dMI.P_8R.\Delta I$

Al mastro M_7 si dispone il FF Ga con $\Delta I.P_8$ onde scrivere il bit gT nel registro.

Si dispone il FF φ con $\Delta I.P_8.M_7$ e si determina la fine dell'istruzione.

Esempio numerico:

da Memoria ÷ ÷ E 521 (15521)

da Registro T 19715

p.d.c. p_3 $1 + 5 = 6$

p_4 $2 + 1 = 3$

p_5 $5 + 7 = 2$ $IRd_1 = 1$

Il riporto $IRd_1 = 1$ viene scritto nel contatore Kr al mastro M_9

p.d.c. p_6 **in U.A.D.**

M_4 $5 + 9 = 4$ $IRd_1 = 1$

M_5 $4 + 1 = 5$

nella rete dei bits e, f

M_4 $e = 1, f = 0$

M_7 si scrive in T il carattere E

M_9 il riporto $IRd_1 = 1$ viene scritto in Kr

M_9 si dispone il FF λ

p.d.c. p_7 **in U.A.D.**

M_4 $0 + 1 = 1$

M_5 $1 + 1 = 2$

M_6 $0 + 1 = 1$ (scompare il riporto IRd_1)

M_7 si scrive in T il carattere 1

nei FF es, fs

M_9 $es = 1, fs = 0$

p.d.c. p_8 **in U.A.D.**

M_4 $0 + 5 = 5$

	nella rete dei bits e, f	
	M_4	e = 0, f = 1
	M_7	si scrive in T il carattere W
Risultato in T		W 236 (35236).

12.3. Caso b

Questo caso non differisce sostanzialmente dal caso a).

Le operazioni sui bits a, b, c, d si arrestano in questo caso al p.d.c. p_6, poichè al p.d.c. p_7 è impedita l'uscita dal registro T di un eventuale carattere posto dopo il carattere contenente il bit gT.

L'eventuale riporto al p.d.c. p_6 non viene conteggiato al p.d.c. p_7, ma al p.d.c. p_8 essendo stato disposto dal segnale di riporto il FF λ al p.d.c. p_6.

Esempio numerico:

da Memoria ÷ ÷ E 521 (15.521)

da Registro T 9715

p.d.c. p_3 $1 + 5 = 6$

p_4 $2 + 1 = 3$

p_5 $5 + 7 = 2$ $IRd_1 = 1$

il riporto $IRd_1 = 1$ viene scritto nel contatore Kr all'M_9.

p.d.c. p_6 **in U.A.D.**

M_4 $5 + 9 = 4$ $IRd_1 = 1$

M_5 $4 + 1 = 5$

nella rete dei bits e, f

M_4 e = 1 ; f = 0

M_7 si scrive in T il carattere E

M_9 il riporto IRd_1 viene scritto in Kr

M_9 si dispone il FF λ

p.d.c. p_7 non si opera alcun carattere

p.d.c. p_8 **in U.A.D.**

M_4 $0 + 5 = 5$

nella rete dei bits e, f

M_4 e = 1 ; f = 1

M_7 si scrive in T il carattere N.

Risultato in T N 236 (25236).

12.4. **Caso c**

Fase β - È vietato operare con caratteri non numerici

p.d.c. p_3 | l'indirizzatore W conta — 1
l'indirizzatore V conta + 1
ai p.d.c. p_4, p_5

p_4 | non conta al p.d.c. p_3

p_5, p_6 | I caratteri letti in Memoria e nel registro T si sommano nella U.A.D. Al mastro M_7 di ogni p.d.c. il risultato viene scritto nel registro T

p_7 l'indirizzatore W conta — 1
l'indirizzatore V conta + 1

Il carattere letto in Memoria non passa attraverso il Complementatore Co per l'assenza del segnale CMU. Il carattere letto nel registro T non viene estratto per la presenza del segnale $\overline{CAU} = \Delta I.\overline{P_8}.\emptyset a$.

Si cancella quindi nel registro T l'eventuale carattere che segue il bit gT, operato al p.d.c. p_6.

p_8 l'indirizzatore W conta — 1
l'indirizzatore V non conta
il carattere # letto in Memoria non viene operato.

Nel registro T si legge il carattere 0 all'M_4 e lo si scrive all'M_7 assieme al bit gT. Al mastro M_7 si dispone il FF φ ed ha termine l'istruzione + IT.

Esempio numerico:

da Memoria # # 1521
da Registro T $\dot{1}6715$

p.d.c. p_3 $1 + 5 = 6$
p_4 $2 + 1 = 3$
p_5 $5 + 7 = 2$ $IRd_1 = 1$

p.d.c. p_6
M_4 $1 + 6 = 7$
M_5 $7 + 1 = 8$

p.d.c. p_7	non si opera alcun carattere
	Si scrive il bit gT nell'indirizzo del carattere letto al p.d.c. p_7.
Risultato in T	$\dot{0}8236$

12.5. **Caso d**

In questo caso si operano 5 caratteri da Memoria e al massimo 5 caratteri da registro T.

Se al p.d.c. p_7 si ha riporto, si dispone il FF Ot che indica « overflow » con $\overline{Db}.IRd.\Delta I.P_7.\overline{I\#M}$, dove è $\overline{I\#M} = \overline{ISM} + dMI$, cioè non si ha # da Memoria. Al p.d.c. p_8 non si conteggia il riporto ottenuto al p.d.c. precedente. L'indirizzatore V non conta e si scrive il bit gT sul carattere scritto al p.d.c. p_7. La sequenza delle operazioni è la stessa esaminata nel caso c.

Esempio numerico:

da Memoria	# 33879	
da Registro T	$\dot{9}7533$	
p.d.c. p_3	9 + 3 = 2	IRd_1 = 1
p_4	7 + 3 = 0	IRd_1 = 1
	0 + 1 = 1	
p_5	8 + 5 = 3	IRd_1 = 1
	3 + 1 = 4	
p_6	3 + 7 = 0	IRd_1 = 1
	0 + 1 = 1	
p_7	3 + 9 = 2	IRd_1 = 1
	2 + 1 = 3	Si dispone Ot
p_8	3 + 0 = 3	
Risultato in T	$\dot{3}1412$.	

12.6. **Istruzione — IT** (# CCCCC T 4)

Decodificati: ΔSXR; ΔI; ΔK

Con la istruzione — IT si sottrae dal contenuto del registro T individuato al 2° p.d.c. della fase α la costante CCCCC estratta da Memoria. Il risultato viene scritto nel registro T.

Vengono esaminati i seguenti casi:

a) la costante in Memoria è formata da 4 caratteri seguiti da due caratteri ÷. Nel registro T deve essere contenuto un numero di 4 o meno di 4 caratteri; se così non fosse si avrebbe un funzionamento non corretto. La costante in Memoria può avere letterale solo il carattere delle migliaia. Il contenuto del registro può essere letterale nelle centinaia e nelle migliaia.

   Memoria ÷ ÷ CCCC

   Registro ˙CCCC

   In questo caso il risultato è un indirizzo di 4 caratteri.

   Il modulo rimane invariato; cioè i bits e ed f delle centinaia, contenuti nel registro T, rimangono invariati. L'indirizzo, nell'ambito del modulo, è la differenza delle due costanti con una aritmetica in base 40.000; cioè sottraendo 1 a 0000 si ottiene 39.999 espresso con 4 cifre in forma letterale.

   La sottrazione viene eseguita sommando al contenuto del T il complemento a 40.000 della costante proveniente da Memoria

b) la costante in Memoria è formata da 4 caratteri seguiti da due caratteri #. Nel registro T è contenuto un numero di 4 o meno di 4 caratteri.

   Le costanti in Memoria e nel registro T non possono essere letterali, altrimenti si avrebbe un funzionamento non corretto.

   Memoria # # CCCC

   Registro ĊCCC

   In questo caso il risultato è un numero di 4 cifre con una aritmetica in base 10.000; cioè sottraendo 1 a 0000 si ottiene 9999; il bit gT viene posto in 5ª posizione, generando nella stessa uno zero.

   La sottrazione viene eseguita sommando al contenuto del T il complemento a 10.000 della costante proveniente da Memoria.

c) La costante in Memoria è formata da 5 caratteri seguiti dal carattere #.

   Nel registro T è contenuto un numero di lunghezza qualsiasi.

Il risultato è un numero di 5 caratteri. Non vi possono essere caratteri letterali nel T e in Memoria.

La sottrazione viene eseguita sommando al contenuto del T il complemento a 100.000 della costante proveniente da Memoria.

Si ha indicazione di « overflow » se non c'è riporto nel p.d.c. p_7.

Esecuzione dell'istruzione.

La fase α dell'istruzione — IT dura come nella istruzione IT fino al p.d.c. P_2.

Le operazioni relative alla fase α sono uguali a quelle descritte nell'istruzione IT.

La fase β dura 6 p.d.c. Nella descrizione della fase β si distinguono i tre casi a, b, c.

12.7. Caso a

Fase β

p.d.c. p_3 l'indirizzatore W conta — 1. L'indirizzatore V non conta.

Si dispone il FF Db con $\Delta I.\Delta R.P_3.M_2$. Il complementatore Co è pronto a complementare a 10 i caratteri uscenti da Memoria fino al primo diverso da zero compreso, ed a 9 i caratteri successivi. È presente il pilotaggio in Memoria e nel registro T e si procede alla lettura dei due primi caratteri. Il complementatore dei bits e ed f lascia invariati i bits provenienti da Memoria (entrambi assenti); si vedrà in seguito come ciò avviene.

I caratteri provenienti da registro T e da Memoria vengono operati nell'U.A.D. e il risultato viene scritto nel registro T.

p_4, p_5 Si opera in modo analogo al p.d.c. p_3 con conta indietro dell'indirizzatore W e conta avanti dell'indirizzatore V.

In p_5 si può avere in uscita del T un carattere letterale; i relativi bits e ed f entrano nel sommatore in

base 4, unitamente ai bits e ed f da Memoria (assenti), e ne escono invariati per cui praticamente si rigenerano nel registro.

p_6 L'indirizzatore W conta — 1. L'indirizzatore V conta + 1.

I caratteri letti in Memoria e nel registro T possono essere alfabetici. I bits a, b, c, d provenienti da Memoria e dal registro vengono operati normalmente; l'eventuale riporto decimale viene staticizzato nel FF λ all'M_9 con $\Delta I.IRd.P_6$.

I bits e ed f provenienti da Memoria passano attraverso il relativo complementatore. Devono essere complementati a 4 se vi è sempre stata complementazione a 10 e il carattere uscito in P_6 è uguale a zero per la parte numerica, altrimenti devono essere complementati a 3.

Si ottiene quindi:
complementazione a 4: $ID1M.IM0.\Delta I$
complementazione a 3: $XU0 = ID1M.\overline{IM0}.\Delta I + IDM.\Delta I$
dove IM0 indica che è uscito zero da Memoria:

$$IM0 = \overline{aMI + cMI + dMI}$$

Si deve però notare che, se non esistono bits e ed f nè da registro nè da Memoria, l'eventuale complementazione a 3, cioè generazione del bit f, deve avvenire solo in questo caso a, e non nel caso b o c; in P_6, non avendo ancora letto il carattere in P_7, non si può sapere se si è nel caso a o in un altro caso.

Si ottiene quindi la logica del complementatore.

$$eMS = fMI.ID1M.IM0.\Delta I + XU0.eMI$$
$$fMS = eMI.ID1M.IM0.\Delta I + XU0.P_6R.XUP.\overline{fMI}$$

Dove il segnale $XUP = eAu + fAu + eMI + fMI$ indica appunto che da registro T o da Memoria è uscita una lettera.
Nel caso sia $\overline{XUP}$, il bit f che si deve generare per la complementazione a 3 viene staticizzato nel FF fs con

$\Delta I.\Delta R.XU_0.\overline{XUP}.P_6.M_9$: verrà utilizzato in P_8 solo se si avrà da Memoria il carattere ÷.

Nei periodi di cifra precedenti a P_6, si può notare che l'unico caso in cui si potrebbe avere uscita di un bit dal complementatore dei bits e ed f è in P_5, in cui si può avere XUP (per la presenza di una lettera nel registro), XU0 e si ha senz'altro $\overline{fMI}$ per questo motivo si è aggiunto P_6R.

Al mastro M_7 si scrive il risultato così ottenuto nel registro T.

p_7 L'indirizzatore W conta — 1. L'indirizzatore V conta + 1.

Il carattere letto in Memoria non passa attraverso il Co per l'assenza del segnale CMU (a causa di ISM).

Il carattere letto nel registro T non viene estratto per la presenza del segnale $\overline{CAU} = \Delta I.\emptyset a.\overline{P_8}$. L'eventuale riporto decimale ottenuto in P_6 è cancellato nel Rr con $ISM.\Delta I.M_6$. Si scrive perciò uno zero nella quinta posizione del registro T.

p_8 L'indirizzatore W conta — 1. L'indirizzatore V non conta ed il suo contenuto viene ridotto di 1 in modo analogo a quanto visto nella IT.

Il carattere ÷ da Memoria non viene operato. Dal registro T viene estratto il carattere scrittosi all'M_7 del p.d.c. P_6. Esso viene modificato nei bits e ed f dall'eventuale riporto decimale ottenuto in P_6 in modo analogo alla + IT, e dall'eventuale bit fs staticizzato nel relativo FF.

Al mastro M_7 si dispone il FF Ga e si scrive il bit gT. Si dispone il FF φ con $\Delta I.P_8.M_7$ e si determina la fine dell'istruzione.

Esempio numerico:

da Memoria ÷ ÷ E 712 (15.712)

da Registro T $\dot{V}$ 314 (34.314)

p.d.c. p_3 8 + 4 = 2 IRd_1 = 1

p_4 8 + 1 + 1 = 0 IRd_2 = 1

p_5 2 + 3 + 1 = 6

p_6 **nell'U.A.D.**

4 + 4 = 8

nella rete dei bits e, f (presente complementazione a 3, XU0)

nel complementatore: ingresso e = 1,f=0 (peso 1)
uscita e = 1,f=1 (peso 2)

nel sommatore: da Memoria peso 2, da accumulatore peso 3, uscita peso 1

p_7 non si opera alcun carattere

p_8 si scrive il bit gT all'indirizzo del carattere letto all'M_7.

Risultato nel T $\dot{H}$602 (18.602).

Altro esempio numerico:

da Memoria ÷ ÷ 1000

da Registro T $\dot{1}$100

p.d.c. p_3 0 + 0 = 0

p_4 0 + 0 = 0

p_5 0 + 1 = 1

p_6 **nell'U.A.D.**

9 + 1 = 0 IRd_1 = 1 si dispone il FF λ

nella rete dei bits e, f

è presente XU0 = $ID1M.\overline{IM0}.\Delta I$

è assente XUP (non esistono bits e ed f)

si dispone il FF fs (complementazione a 3 di un « peso » 0)

nel registro T si scrive uno zero

p_7 non si opera alcun carattere

p_8 si scrive il bit gT, inoltre nel sommatore in base 4 si sommano l'fs (peso 3) e il riporto in λ (peso 1); si scrive uno zero.

Risultato $\dot{0}$100.

Altro esempio numerico:

da Memoria ÷ ÷ 1000

da Registro T $\dot{0}$000

p.d.c. p_3, p_4, p_5

0 + 0 = 0

p.d.c. p_6 **nell'U.A.D.**

9 + 0 = 9

nella rete dei bits e, f

si dispone il FF fs, nel registro T si scrive il solo 9

p_7 non si opera alcun carattere

p_8 **nella rete dei bits e, f**

Si somma il peso 3 contenuto nell'fs con il peso 0 del registro;
si aggiunge perciò il bit f al 9 già scritto nel registro; si scrive il carattere '.

Risultato nel registro '000 (39.000).

12.8. **Caso b**

Fase β

p.d.c. p_3, p_4, p_5, p_6 si opera in modo analogo al caso a; si ricorda che in p_5 e in p_6 non si possono avere caratteri letterali

p.d.c. p_7 analogo al caso a

p.d.c. p_8 l'indirizzatore W conta — 1; V non conta; il carattere letto in Memoria non viene operato; nel registro T si legge il carattere 0 e lo si riscrive all'M_7 assieme al bit gT.

Esempio numerico:

da Memoria # # 1000

da Registro T $\dot{1}$100

p.d.c. p_3, p_4 0 + 0 = 0

p_5 0 + 1 = 1

p_6 **nell'U.A.D.**

9 + 1 = 0 IRd = 1. Si dispone il FF λ

nella rete dei bits e, f

si dispone il FF fs

nel registro si scrive 0

p_7 non si opera alcun carattere; nel registro si scrive 0

p_8 si scrive il bit gT; essendovi # da Memoria il sommatore in base 4 non ha alcun ingresso.

Risultato $\dot{0}0100$.

Altro esempio numerico:

da Memoria # # 1000

da Registro T $\dot{0}000$

p.d.c. p_3, p_4, p_5, p_6, p_7 come nell'identico esempio al caso a

p_8 si scrive il bit gT; il sommatore in base 4 non ha alcun ingresso.

Risultato $\dot{0}9000$.

12.9. **Caso c**

In questo caso si operano 5 caratteri da Memoria e al massimo 5 caratteri da registro T.

Se al p.d.c. P_7 non si ha riporto si dispone il FF Ot con $Db.\overline{IRd}.\Delta I.P_7.\overline{I\#M}$; $\overline{I\#M} = ISM + dMI$.

Al p.d.c. P_8 non si conteggia il riporto ottenuto al p.d.c. precedente.

L'indirizzatore V non conta e si scrive il bit gT sul carattere scritto al p.d.c. P_7. La sequenza delle operazioni è la stessa esaminata nel caso b.

Esempio numerico:

da Memoria # 12345

da Registro T $\dot{0}2345$

p.d.c. p_3 $5 + 5 = 0$ $IRd_1 = 1$

p_4 $5 + 4 + 1 = 0$ $IRd_2 = 1$

p_5 $6 + 3 + 1 = 0$ $IRd_2 = 1$

p_6 **nell'U.A.D.**

$7 + 2 + 1 = 0$ $IRd_2 = 1$. Si dispone il FF λ

nella rete dei bits e, f

si dispone il FF fs

p_7 8 + 0 + 1 = 9 si dispone il FF Ot

p_8 non si utilizzano i FF λ ed fs; nel registro T si riscrive il 9, aggiungendovi il bit gT.

Risultato 90.000.

12.10. **Istruzione ITT** (X Tb IIII Ta #)

Decodificati ΔSΣR; ΔK.

Con la istruzione ITT si sottrae il contenuto delle prime quattro posizioni del registro Tb, individuato al p.d.c. p_7 della fase α, all'indirizzo di Memoria definito ai p.d.c. p_3, p_4, p_5, p_6 della fase α. Al p.d.c. p_2 si individua l'indirizzo del registro Ta modificatore. Il risultato viene scritto in Tb. La quinta posizione di Tb rimane invariata.

L'operazione avviene solo fra numeri non segnati.

Preparazione ed esecuzione dell'istruzione ITT

Fase α

p.d.c. p_0 Si trasferisce il contenuto del registro I all'indirizzatore W. L'indirizzatore W conta + 10.

p_1 L'indirizzatore W conta — 1
Si legge in Memoria il carattere di funzione e lo si staticizza nei FF F.

p_2 Si azzera l'indirizzatore V
L'indirizzatore W conta — 1. Si legge in Memoria e si scrive nell'indirizzatore V l'indirizzo del registro T modificatore.

p_3, p_4 Ai p.d.c. p_3, p_4, p_5, p_6 si legge in Memoria e nel registro T alle posizioni individuate dalle conte degli indirizzatori W (indietro) e V (avanti)

p_5, p_6 I caratteri letti passano inalterati attraverso i rispettivi complementatori Co e Ca. Essi vengono operati nella U.A.D. e i risultati vengono scritti ai singoli p.d.c. nel registro 0.

p_7 L'indirizzatore W conta — 1

Il carattere letto in Memoria viene trasferito ai FF Lu.

p_8 L'indirizzatore W conta — 1

L'indirizzatore V viene azzerato.

Si dispone il FF η con $\alpha.P_8\Delta S\Sigma R.M_2$ e viene a mancare il pilotaggio in Memoria durante la fase β.

Si trasferisce all'indirizzatore V il carattere operato al p.d.c. precedente in modo analogo alla MT; è XWC = $\alpha.P_8.\Delta S\Sigma R$. Si predispone il pilotaggio del registro T selezionato. Si ripone il FF Kz. Si azzera il contatore Cp. Si ripone il FF α e si dispone il FF βw.

Fase β ed η

È analoga alla fase η già vista nella MA; inoltre:

a) L'indirizzatore W non conta con $\eta.P_1$; la conta è inibita con $\eta.\Delta S\Sigma R$

b) Lo strobe del registro T non è inibito essendo presente ΔΣR; si legge quindi nel registro T; il carattere letto viene complementato a 10 fino alla prima cifra diversa da zero, a 9 per le successive; è infatti DA = $\beta r.\Delta S\Sigma R$

Non avendosi un complementatore dei bit e, f da accumulatore non si può operare correttamente in termini di indirizzi letterali di Memoria.

Il risultato ottenuto nell'U.A.D. viene scritto nel registro T.

13. ISTRUZIONI DI CONFRONTO

13.1. Generalità

Con le istruzioni di confronto si operano i confronti seguenti:

1. Caratteri provenienti da Memoria con caratteri provenienti da accumulatore (istruzione CMA)
2. Caratteri provenienti da Memoria con caratteri provenienti da un registro T (istruzione CMT)
3. Caratteri provenienti da una zona di Memoria con caratteri provenienti da un'altra zona di Memoria (istruzione CMM)
4. Caratteri di una istruzione con caratteri provenienti da un registro T (istruzione CIT).

Il confronto è realizzato da una apposita rete logica. Il risultato del confronto è espresso dallo stato dei FF C $<$, C $\neq$.

Possono darsi i seguenti casi:

1° termine di confronto $=$ al 2° termine di confronto
(FF C$\neq$ riposto)

1° termine di confronto $\neq$ dal 2° termine di confronto
(FF C$\neq$ disposto)

1° termine di confronto $>$ del 2° termine di confronto
(FF C$\neq$ disposto e FF C$<$ riposto)

1° termine di confronto $<$ del 2° termine di confronto
(FF C$\neq$ disposto e FF C$<$ disposto)

Il 1° termine di confronto è rappresentato da:

1) Una parola contenuta in Memoria nelle istruzioni CMA, CMT
2) Una costante espressa dalla istruzione nelle istruzioni CIT
3) Una parola contenuta in Memoria col primo carattere all'indirizzo espresso dall'istruzione, nelle CMM.

   Questa parola viene confrontata con quella individuata in Memoria dall'indirizzo specificato da una precedente istruzione PUM; come in tutte le ΔMM ciò che esce dalla Memoria all'indirizzo specificato della PUM va al complementatore di accumulatore.

Il confronto viene effettuato carattere per carattere ed è decisivo il confronto dell'ultimo carattere.

Si ricorda che il confrontatore dei caratteri provenienti dalla Memoria (BMI) e dall'accumulatore (BCa) fornisce i seguenti segnali:

IM $\neq$ α indica che i due caratteri sono diversi per la parte alfabetica (cioè per i bits e ed f)

IM $\neq$ n indica che i due caratteri sono diversi per la parte numerica (cioè per i bits a, b, c, d)

XCA indica che il carattere di Memoria è minore (o uguale) al carattere proveniente dal complementatore di accumulatore.

XCB indica che il carattere di Memoria è maggiore (o uguale) al carattere proveniente dal complementatore di accumulatore.

Per maggiori dettagli si può vedere la « Descrizione logica degli organi », pag. 150 e seguenti.

La logica del confronto opera per mezzo dei FF C $<$, C $\neq$ nel caso in cui i due termini del confronto non siano segnati od abbiano segni uguali e per mezzo dei FF C $<$, C $\neq$, Cs, nel caso in cui i due termini del confronto siano segnati e con segni diversi.

1. Nel primo caso il FF Cs rimane riposto per tutta la durata del confronto.

   Il FF C $<$ riceve invece ad ogni p.d.c. sulla maschera diretta il segnale XCA (nella CMA solo se c'è IA+, sempre nelle altre istruzioni), o il segnale XCB (se nella CMA si ha IA—).

   Il FF C $\neq$, che viene riposto in $\beta.P_1.M_6$ nelle CMA, CMT, CMM e in $\beta.P_3.M_6$ nella CIT, viene disposto al mastro M_2 del p.d.c. successivo a quello in cui sono stati confrontati due caratteri diversi.

2. Nel secondo caso, si dispone all'M_9 del p.d.c. P_1 il FF Cs. Al mastro M_2 del p.d.c. P_2 si dispone il FF C $\neq$, mentre il FF C $<$ si dispone se il segno del 1° termine del confronto è —, e si ripone se è +.

   Il FF Cs dispone in $P_2.M_7$ il FF φ e finisce immediatamente l'istruzione di confronto.

Si devono qui notare due casi particolari:

a) Nella CMM con parole entrambe segnate negative il risultato del confronto tiene conto solo del valore assoluto delle parole stesse

b) Nella CIT non si tiene conto nel confronto dei caratteri sostituito con un # nell'istruzione.

13.2. **Istruzione CMM** (00 IIII T/)

Decodificato ΔSEB

Si confronta la parola contenuta in Memoria a partire dall'indirizzo IIII modificato dal registro T e di lunghezza specificata in una precedente istruzione PUM, con una parola contenuta in Memoria a partire dall'indirizzo specificato dalla precedente istruzione PUM. Le due parole hanno la stessa lunghezza e debbono essere entrambe segnate o senza segno.

La istruzione CMM impegna entrambi i canali (interno e esterno) della U.C.

All'inizio della fase α preparatoria l'indirizzatore Y contiene l'indirizzo iniziale della parola che costituisce il 2° termine di confronto. Il FF βy è disposto.

Preparazione ed esecuzione della istruzione CMM

Fase α

p.d.c. p_0 si azzera l'indirizzatore W
Si trasferisce all'indirizzatore W il contenuto del registro I. L'indirizzatore W conta + 10

p_1 l'indirizzatore W conta — 1
Si legge in Memoria e si trasferisce il carattere letto nei FF di funzione F

p_2 l'indirizzatore W conta — 1

Si legge in Memoria e si scrive nell'indirizzatore V l'indirizzo del registro T modificatore.

p_3 p_4 p_5 p_6 — Si trasferiscono ai singoli p.d.c. i caratteri letti in Memoria e modificati dal contenuto del registro T. L'indirizzatore W conta — 1 ad ogni p.d.c. L'indirizzatore V non conta al p. d. c. p_3 e conta + 1 ai p.d.c. p_4, p_5, p_6

p_7 p_8 — Si trasferiscono le cifre 0 ai p.d.c. p_7 e p_8 rispettivamente in Lu e Ld. In questo modo non si altera il contenuto dei FF Lu ed Ld che contengono la lunghezza dell'operando specificata dalla istruzione PUM precedente.

Fase β

p.d.c. p_0 M_0 si azzera l'indirizzatore W

M_3 si ripone il FF α_r

M_3 si dispone il FF βr

M_4 si ripone il FF Iy

M_4 si trasferisce all'indirizzatore W il contenuto del registro 0

M_5 si dispone il FF Kz

M_8 si dispone il FF Ce. Se supponiamo pari il contenuto in Y, si ha il segnale Ce.IYP

M_9 si ripone il FF p_0

M_9 il contatore Cp conta + 1. Il decodificatore di p-pulsi fornisce il segnale p_1

p.d.c. p_1 M_0 l'indirizzatore W conta — 1
l'indirizzatore Y conta — 1

M_0 si dispone il FF Zc

M_3 si dispone il FF Zr

M_0 sono presenti i segnali di consenso al pilotaggio pMp, pMd

M_4 si legge in Mp e si staticizza il carattere letto in Up

p_1 M_4 si legge in Md e si staticizza il carattere letto in Ud

M_6 si ripone il FF C $\neq$

M_7 si rigenera in Memoria Md il carattere letto al mastro M_4

M_7 si rigenera in Memoria Mp il carattere letto al mastro M_4

M_9 il contatore Cp conta + 1. Il decodificatore di p-pulsi fornisce il segnale p_2

p.d.c. p_2 M_0 l'indirizzatore W conta — 1
l'indirizzatore Y conta — 1

M_0 si ripone il FF Zc

M_2 si dispone o ripone in base alle indicazioni della rete logica il FF C $<$; si dispone in base alle indicazioni della rete logica il FF C $\neq$.

Il carattere letto precedentemente nella Memoria Up e staticizzato fino al mastro M_3 successivo nei FF Up, viene direttamente introdotto nella rete logica di confronto dall'uscita dei FF Up (BMI).

Il carattere letto precedentemente nella Memoria Ud e staticizzato fino al mastro M_3 successivo nei FF Ud, viene portato alla rete logica di confronto prelevando le uscite negate dei FF Ud e invertendole (BME)

M_3 si ripone il FF Zr

M_3 si ripongono i FF Ud, Up

M_4 si legge in Memoria Mp e si staticizza il carattere letto nei FF Up

M_4 si legge in Memoria Md e si staticizza il carattere letto nei FF Ud

M_7 si rigenera in Memoria Up il carattere letto al mastro M_4

M_7 si rigenera in Memoria Ud il carattere letto al mastro M_4

M_9 il contatore Cp conta + 1. Il decodificatore di p-pulsi fornisce il segnale p_3.

Nei p.d.c. seguenti le operazioni si ripetono eguali a quelle descritte al p.d.c. p_2. Ad ogni p.d.c. il FF C < viene disposto o riposto, mentre il FF C ≠ rimane disposto dopo il primo confronto di due caratteri ≠.

Quando si opera l'ultimo carattere compare al mastro M_5 il segnale ∅c. Al mastro M_7 si dispone il FF φ.

Al mastro M_7 si ripone il FF Kz e all'M_9 si azzera il contatore Cp e si dispongono i FF p_0 ed α.

L'UC è pronta a preparare una successiva istruzione.

Il risultato dell'ultimo confronto viene registrato al mastro M_2 del p.d.c. successivo a quello in cui i caratteri sono stati confrontati.

Il risultato del confronto rimane staticizzato nei FF C < e C ≠.

La fine degli operandi può essere definita oltre che dal confrontatore di lunghezza Cl, dal confrontatore di ricerca Cr e dalla rete logica che fornisce i segnali Isα, I∅I se nel FF Lu è stato introdotto al p.d.c. p_7 della fase α uno dei caratteri P. Q. Y. Z.

Come già illustrato a proposito della istruzione MA, se in Lu sono stati posti i caratteri Z o Y si ottengono i segnali Isα e ∅c quando si legge in Memoria un carattere non numerico; se in Lu sono stati posti i caratteri P o Q si ottengono i segnali I∅I, ∅c, quando si legge in Memoria il carattere chiave posto in Ld. In tutti e quattro i casi non si effettua il confronto dell'ultimo carattere letto poichè i FF Up e Ud vengono riposti al mastro M_0, mentre il risultato del confronto viene staticizzato al mastro M_2. In particolare la presenza dei caratteri Q e Z è accompagnata da una fase η che permette di staticizzare nel registro T_9 l'indirizzo dell'ultimo carattere operato della parola che costituisce il 1° termine di confronto.

Durante la fase η, che segue il segnale ∅c, le operazioni si svolgono come descritto a proposito della istruzione MA. L'indirizzatore Y non è interessato alle operazioni, la sua conta è impedita dal riporsi del FF Ce, che assieme al segnale η toglie il pilotaggio alle Memorie Mp, Md. Al termine della fase η si dispone il FF φ che determina la fine delle operazioni relative all'istruzione CMM, mentre si ripone anche il FF βy, liberando il canale esterno della U.C.

13.3. **Istruzione CIT** (# CCCCC T5)

Decodificato ΔSUB

Con la istruzione CIT si confronta la costante CCCCC posta in Memoria col contenuto del registro T individuato al secondo p.d.c. Gli eventuali caratteri # posti nella costante CCCCC non danno luogo al confronto nei corrispondenti p.d.c., poichè il carattere # letto in Memoria impedisce che si disponga o riponga il FF C $<$.

La fase α della istruzione CIT dura 3 p.d.c., ad essa segue la fase β della durata di 6 p.d.c.

Preparazione ed esecuzione della istruzione CIT

Fase α

p.d.c. p_0 si azzera l'indirizzatore W, vi si trasferisce il contenuto del registro I. L'indirizzatore W conta +10.

p_1 L'indirizzatore W conta — 1
Si legge in Memoria il carattere di funzione che viene staticizzato nei FF F

p_2 si azzera l'indirizzatore V
L'indirizzatore W conta — 1.

Si legge in Memoria e si scrive nell'indirizzatore V l'indirizzo del registro T che contiene il secondo termine di confronto.

Fase β

p.d.c. p_3 L'indirizzatore W conta — 1
L'indirizzatore V non conta.
Viene riposto il FF C $\neq$.

È presente il segnale di consenso al pilotaggio del registro T. Si legge in Memoria e in registro T.

I caratteri letti vengono staticizzati rispettivamente nei FF Up (Ud) e Au. (I caratteri letti vengono rigenerati in Memoria e nel registro T).

p_4 L'indirizzatore W conta — 1
L'indirizzatore V conta + 1.

Si dispongono (o ripongono) i FF C< e C≠, staticizzando il risultato del confronto relativo ai caratteri operati al p.d.c. precedente.

Si ripongono i FF Up (Ud) e Au. Si legge in Memoria e in registro T staticizzando i caratteri letti nei FF Ud (Up) ed Au. (I caratteri letti vengono rigenerati in Memoria e in registro T).

p_5
p_6 } si opera come al p.d.c. p_4
p_7

p_8 L'indirizzatore W conta — 1
L'indirizzatore V non conta.

Si modifica lo stato del FF C< in relazione al risultato del confronto dei caratteri operati al p.d.c. p_7. (Il FF C≠ si dispone al p.d.c. seguente a quello in cui il confronto ha dato indicazione che i due termini di confronto sono ≠).

Lo stato del FF C≠ non viene ulteriormente modificato se già disposto.

Si legge in Memoria il carattere # e nel registro T il carattere operato al p.d.c. p_7 se questo non conteneva il bit gT. I caratteri letti vengono rigenerati.

La presenza del carattere # nei FF Ud (Up) impedisce la modifica dello stato dei FF C< e C≠ e non ha luogo alcun confronto in questo p.d.c.

Si dispone il FF φ che determina la fine delle operazioni relative all'istruzione CIT.

14. ALTRE ISTRUZIONI

14.1. Istruzione DA (XXXX II TA)

Decodificato ΔAUP

L'istruzione DA precede un'istruzione con la quale si operi sull'accumulatore. Con essa si introduce nel registro **P** il numero espresso dai caratteri operati ai p.d.c. p_3, p_4 della fase α. Tale numero indica l'indirizzo di accumulatore che verrà individuato per primo nell'istruzione successiva.

L'istruzione DA consta di una fase α della durata di nove p.d.c. (p_0 - p_8) e di una fase β che termina al p.d.c. p_0.

Preparazione ed esecuzione dell'istruzione DA

Fase α

p.d.c. p_0 M_0 il FF α è disposto
- M_0 si cancella il contenuto dell'indirizzatore W
- M_2 si dispone il FF Dw
- M_3 si dispone il FF αr
- M_3 si ripone il FF Sm
- M_4 si trasferisce il contenuto del registro I in W e si predispone la maschera del FF Zc (esempio $\overline{Ce}$.IWD)
- M_5 si dispone il FF Kz
- M_7 l'indirizzatore W conta + 10
- M_8 si ripone il FF Dw
- M_9 il contatore Cp conta + 1. Il decodificatore di p-pulsi fornisce il segnale p_1
- M_9 si ripone il FF α_{37}
- M_9 si ripone il FF p_0

p.d.c. p_1 M_0 l'indirizzatore W conta — 1 (IWD in IWP)
- M_0 si dispone il FF Zc (il segnale IWP non ha effetto perchè arriva in maschera)
- M_0 è presente il segnale pMp per il pilotaggio della Memoria Mp

p_1 M_2 si cancella il contenuto dei FF F

M_3 si dispone il FF Zr

M_3 si ripongono i FF Up

M_3 si ripongono i FF Rm

M_4 si legge in Mp. (Il carattere letto viene rigenerato al successivo mastro M_7)

M_4 si staticizza nei FF Up il carattere letto in Mp

M_4 si trascrive nei FF F il contenuto dei FF Up

M_5 i FF Rm si predispongono sul valore 1

p.d.c. p_2 M_0 l'indirizzatore W conta — 1 (IWP in IWD)

M_0 scompare il segnale pMp e compare il segnale pMd

M_0 si ripone il FF Zc

M_0 si azzera l'indirizzatore V e si ripone il FF fV

M_3 si ripone il FF Zr

M_3 si ripongono i FF Ud di uscita da Md

M_4 si legge in Md (il carattere letto viene rigenerato al mastro M_7 successivo)

M_4 si staticizza nei FF Ud il carattere letto in Md

M_7 il contenuto dell'indirizzatore W passa nel registro I

M_8 si trasferisce nell'indirizzatore V il contenuto dei FF Ud. Il carattere operato rimane inalterato nel passaggio attraverso Co per la presenza del segnale $\overline{IDM}$ e nel passaggio attraverso l'U.A.D. essendo Rm = 1 e mancando il pilotaggio in accumulatore e registri T.

I bits a, b, c, d, del carattere letto vanno a staticizzarsi in Vd; una rete logica dei bits e ed f scrive in Vu la cifra 0 o 5, mentre il FF fv disponendosi o rimanendo riposto, individua il pilotaggio PT_I, PT_{II} dei registri T per il p.d.c. seguente

M_9 si dispone il FF α_{37}. È presente il segnale PT_I PT_{II} per il pilotaggio delle Memorie T_I o T_{II}

p.d.c. p_3 M_0 l'indirizzatore W conta — 1 (IWD in IWP)

M_0 scompare il segnale pMd e compare il segnale pMp

M_0 si dispone il FF Zc

M_3 si dispone il FF Zr

M_4 si legge il carattere in Mp (il carattere letto viene rigenerato al mastro M_7 successivo)

M_4 si staticizza il carattere letto in Mp nei FF Up

M_4 si legge nel registro T (il carattere letto viene rigenerato al mastro M_7 successivo)

M_4 si staticizza nei FF Au il carattere letto in T

M_4 il carattere letto in Mp attraversa il complementatore Co inalterato per la presenza del segnale $\overline{IDM}$; il carattere uscente dal registro T attraversa il complementatore Ca inalterato per la presenza del segnale $\overline{IDA}$. Essendo il contenuto dei FF Rm = 1, l'U.A.D. esegue la somma dei due caratteri estratti. Il risultato rimane disponibile nella U.A.D.

M_7 il risultato viene scritto nel registro P unità (AJP $\equiv$ ΔAUP.P_3)

p.d.c. p_4 M_0 l'indirizzatore V conta + 1

M_0 l'indirizzatore W conta — 1 (IWP in IWD)

M_0 scompare il segnale pMp e compare il segnale pMd

M_0 si ripone il FF Zc

M_3 si ripone il FF Zr

M_4 si ripongono i FF di uscita da Md

M_4 si legge in Md. Il carattere letto viene rigenerato al mastro M_7 successivo

M_4 si staticizza nei FF Ud il carattere letto in Md

M_4 si legge nel registro T. Il carattere letto viene rigenerato al mastro M_7 successivo

M_4 si staticizza nei FF Au il carattere letto in T

p_4 M_4 si esegue in U.A.D. la somma dei due caratteri in modo analogo a quanto avvenuto al p.d.c. precedente

M_7 si scrive il risultato nel registro P decine.

I caratteri corrispondenti ai p.d.c. p_5, p_6, p_7, p_8 non hanno interesse e possono essere espressi col carattere #.

Col p.d.c. p_8 termina la fase α.

Al mastro M_8 si ripone il FF Kz

Al mastro M_9 si azzera il contatore Cp

Al mastro M_9 si dispone il FF Po

Al mastro M_9 si ripone il FF α

Al mastro M_9 si dispone il FF βw.

Fase β

p.d.c. p_0 M_0 il FF βw è disposto

M_0 si azzera l'indirizzatore W

M_3 si ripone il FF α_r e scompare il segnale pMd

M_3 si dispone il FF βr

M_5 si dispone il FF Kz

M_7 si dispone il FF φ con $\beta w.P_0.\Delta AUP$

M_7 si ripone il FF Kz

M_9 si ripone il FF Po

M_9 si azzera il contatore Cp

M_9 si dispone il FF α.

Disponendosi il FF α per la presenza del segnale di fine φ, si può iniziare la preparazione dell'istruzione successiva alla DA.

14.2. **Istruzione FAM** (X X IIII Tδ)

Decodificato ΔAER

La istruzione FAM permette di scrivere all'indirizzo di Memoria IIII modificato da T (espresso dalla istruzione stessa) un numero che indica la lunghezza della parola contenuta in Accumulatore e com-

presa fra l'indirizzo specificato da una precedente istruzione DA e e il bit gA posto sull'ultimo carattere dell'operando in Accumulatore. Si distinguono due casi:

a) il carattere letto al p.d.c. p_8 della fase α è un qualsiasi carattere col bit f = 0. In questo caso vengono conteggiate solo le cifre significative della parola letta in Accumulatore

b) il carattere letto al p.d.c. p_7 della fase α è zero e quello letto al p.d.c. p_8 è il carattere (punto), cioè con il solo bit f = 1. In questo caso la lunghezza della parola letta in Accumulatore viene valutata tenendo conto di tutti i caratteri compresi fra l'indirizzo iniziale e quello relativo al bit gA che indica la fine della parola in Accumulatore.

Preparazione ed esecuzione dell'istruzione FAM

Fase α

p.d.c. p_0 si azzera l'indirizzatore W e vi si trasferisce il contenuto del registro I. L'indirizzatore W conta + 10

p_1 l'indirizzatore W conta — 1, il carattere di funzione viene scritto nei FF F.
Si azzerano i FF Lu, Ld.

p_2 l'indirizzatore W conta — 1
l'indirizzatore V viene azzerato.

Si scrive nell'indirizzatore V l'indirizzo del registro T modificatore.

p_3, p_4 — si estraggono da Memoria nei singoli p.d.c. i caratteri dell'indirizzo IIII.

p_5, p_6 — Essi vengono modificati dal contenuto del registro T e staticizzati nel registro 0.

p_7 — i caratteri letti in Memoria ai p.d.c. p_7, p_8 vengono trasferiti ai FF Lu, Ld

p_8 — Al mastro M_0 del p.d.c. p_8 viene azzerato l'indirizzatore V e al mastro M_4 viene trasferito in V il contenuto del registro JP.

Col p.d.c. p_8 termina la fase α dell'istruzione FAM ed inizia la fase esecutiva β. Le operazioni relative alla fase β sono diverse a seconda che il contenuto dei FF Lu, Ld corrisponda alle situazioni a) e b) precedentemente esaminate.

Caso a)

Il carattere contenuto in Ld ha il bit f = 0.

In questo caso si tiene conto nella valutazione della lunghezza della parola in Accumulatore delle sole cifre significative, ciò significa che la parola in Accumulatore è un numero. Questo numero può essere in forma normale o in complemento. Se è in complemento è disposto il FF Cn e i caratteri uscenti da Accumulatore vengono di nuovo complementati prima che l'apposita rete logica provveda a valutare se la cifra letta è significativa ($DA = \Delta AER.\overline{Ap}.XAZ$).

Si ha complementazione a 10 finchè non esce da Accumulatore una cifra $\neq 0$ che disponga il FF A $\neq$.

Il FF A $\neq$ si attiva al mastro M_3 col primo carattere $\neq 0$; poichè i caratteri vengono letti al mastro M_4, il FF A $\neq$ si dispone nel periodo di cifra seguente a quello in cui è stato letto il primo carattere $\neq 0$.

Nell'esecuzione dell'istruzione il contatore Cp conta + 1 ad ogni periodo di cifra. In corrispondenza ad ogni carattere operato $\neq 0$ è presente il segnale IAu $\neq$.

Al mastro M_7 di un generico periodo di cifra con IAu $\neq$ si azzerano i FF Lu, Ld e al mastro M_9 dello stesso periodo di cifra si trasferisce in Lu, Ld il contenuto del contatore Cp. Dato che nelle operazioni aritmetiche, si parte dall'esame delle cifre meno significative, tutte le cifre 0 che seguono l'ultima $\neq 0$ non dovranno essere registrate in Lu, Ld. Ciò è realizzato dalla rete logica, poichè la cancellazione e la riscrizione in Lu, Ld è possibile solo con la lettura di un carattere $\neq 0$. Se l'ultimo carattere letto in Accumulatore è $\neq 0$ l'azzeramento dei FF Lu, Ld avviene attraverso la porta $gAu.\overline{fLd}.\Delta AER$, anzichè attraverso la porta $\overline{\varnothing a}.\overline{fLd}.\Delta AER$ come avviene nei p.d.c. precedenti.

Fase β

p.d.c. p_0 M_0 si azzera l'indirizzatore W

M_3 si ripone il FF αr

M_3 si dispone il FF βr

M_4 si trasferisce all'indirizzatore W il contenuto del registro 0

M_5 si ripone il FF ∅a

M_5 si dispone il FF Kz

M_9 il contatore Cp conta + 1. Il decodificatore di p-pulsi fornisce il segnale p_1

p.d.c. p_1 M_0 l'indirizzatore W non conta

La sua conta è impedita dalla presenza del segnale $\overline{\emptyset a}$, che toglie anche il pilotaggio in Memoria.

L'indirizzatore V non conta

M_0 è presente il segnale PA di consenso al pilotaggio in Accumulatore

M_3 si ripongono i FF Au

M_4 si legge in Accumulatore in corrispondenza all'indirizzo contenuto nell'indirizzatore V

M_7 la cifra letta all'M_4 viene rigenerata in Accumulatore

M_7 si cancellano i FF Lu, Ld se presente il segnale IAu ≠

M_9 si scrive nel FF Lu la prima conta di Cp

M_9 il contatore Cp conta + 1. Il decodificatore di p-pulsi fornisce il segnale p_2.

Nei p.d.c. seguenti si ripetono le stesse operazioni con conta in avanti dell'indirizzatore V.

Nell'ultimo p.d.c. all'uscita del bit gA (M_4) seguono le seguenti operazioni:

M_4 si ripone il FF Kz con βw.gAu.ΔAER

M_6 si dispone il FF ∅a

M_7 eventualmente si cancellano i FF Lu, Ld

M_9 si trasferisce in Lu, Ld il contenuto del contatore Cp

M_9 si azzera il contatore Cp

M_9 si dispone il FF Po.

Il contatore Cp riprende la conta partendo da 0

p.d.c. p_0 M_0 si azzera l'indirizzatore W

M_4 si trasferisce all'indirizzatore W il contenuto del registro 0

M_5 si dispone il FF Kz

M_9 il contatore Cp conta + 1

Il decodificatore di p-pulsi fornisce il segnale p_1

p.d.c. p_1 M_0 l'indirizzatore W conta — 1

È presente il consenso al pilotaggio in Memoria e manca quello al pilotaggio in Accumulatore

M_4 si legge in Memoria all'indirizzo indicato dall'indirizzatore W cancellando il contenuto della Memoria

M_7 si trasferisce la cifra contenuta in Lu all'indirizzo di Memoria indicato dall'indirizzatore W

M_9 il contatore Cp conta + 1, il decodificatore di p-pulsi fornisce il segnale p_2

p.d.c. p_2 M_0 l'indirizzatore W conta — 1

M_4 si cancella la posizione di Memoria corrispondente all'indirizzo contenuto in W

M_7 si trasferisce in Memoria la cifra contenuta nei FF Ld

M_7 si dispone il FF φ

M_7 si ripone il FF Kz

M_9 si azzera il contatore Cp

M_9 si dispone il FF Po

M_9 si dispone il FF α.

Disponendosi il FF α si passa alla preparazione di una successiva istruzione.

Caso b)

Se in Lu è stato introdotto il carattere 0 al p.d.c. p_7 della fase α e in Ld il carattere . al p.d.c. p_8, il trasferimento dal contatore Cp ai FF Lu, Ld avviene con l'estrazione da Accumulatore dell'ultimo carattere. Il numero che viene quindi a staticizzarsi nei FF Lu, Ld tiene conto di tutti i caratteri operati; il trasferimento del contenuto di Lu, Ld in Memoria avviene come nel caso precedente.

14.3. **Istruzione FTM** (LT.IIII $T_m\Theta$)

Decodificati ΔTER, ΔAER

Da notare che anche nella FTM, pur essendovi ΔT e non ΔA, si ha il segnale ΔAER; è infatti ΔAER = ΔA.ΔE.ΔR + ΔT.ΔE.ΔR.

L'istruzione FTM, in modo analogo alla FAM, scrive in Memoria la lunghezza della parola contenuta nel registro To.

Anche per questa istruzione si hanno i due casi a) e b).

Fase α

È analoga alla fase α della MT.

L'unica differenza è che il carattere letto in P_8 va nel registro lunghezze decine, oltre che nel registro lunghezze unità, per la presenza di ΔAER.

Fase β

È uguale a quella della FAM.

L'unica differenza è che si ha il pilotaggio dei registri (invece dell'Accumulatore) con XAF = $\Delta T.\beta r.\overline{\emptyset a}$.

14.4. **Istruzione RIa** (LL IIII T2)

Decodificato ΔSEP; ΔRI

Con la istruzione RIa si ricerca in Memoria partendo dall'indirizzo IIII modificato dal contenuto di T e procedendo in avanti, l'indirizzo di un particolare carattere individuato dal contenuto dei FF Lu, Ld. La ricerca di questo carattere avviene confrontando i

bits di ciascun carattere estratto da Memoria con quelli del carattere posto in Ld al p.d.c. p_8 della fase α. Il confronto avviene nel confrontatore di ricerca Cr ed è condizionato dal carattere posto in Lu al p.d.c. p_7 della fase α. Il condizionamento della ricerca è tale per cui vengono interessati al confronto non tutti i bits dei due caratteri, ma solo quelli che nel carattere condizionatore sono eguali ad 1. Ponendo in particolare in Lu il carattere Q si ricerca in Memoria il carattere posto in Ld.

Quando la ricerca è terminata si inizia una fase η della durata di 5 p.d.c. durante la quale si trasferisce nel registro T_9 l'indirizzo ricercato. La istruzione RIa dura complessivamente 15 + LR p.d.c., essendo LR il numero di p.d.c. necessario per individuare il carattere ricercato.

Preparazione ed esecuzione dell'istruzione RIa

Fase α

p.d.c. p_0 si azzera l'indirizzatore W e vi si trasferisce il contenuto del registro I. L'indirizzatore W conta + 10

p_1 l'indirizzatore W conta — 1

Il carattere di funzione viene letto in Memoria e staticizzato nei FF F

p_2 l'indirizzatore W conta — 1

Si scrive nell'indirizzatore V l'indirizzo del registro T modificatore

p_3, p_4 si estraggono da Memoria nei singoli p.d.c. i caratteri dell'indirizzo IIII.

p_5, p_6 Essi vengono modificati dal contenuto del registro T e staticizzati nel registro 0

p_7 si legge in Memoria il carattere condizionatore della ricerca e lo si trasferisce nei FF Lu

p_8 si legge in Memoria il carattere di confronto e lo si trasferisce nei FF Ld.
Si cancella il contenuto dell'indirizzatore V e vi si

introduce l'indirizzo del registro T_9. (Ciò è ottenuto disponendo in α, p_8 M_1 i FF dell'indirizzatore Vd relativi ai bits c e d)

Fase β

p.d.c. p_0 M_0 si azzera l'indirizzatore W

M_0 è presente il segnale di consenso al pilotaggio in Memoria pMp (pMd)

M_3 si ripone il FF αr e scompare il segnale di pilotaggio pMd (pMp)

M_3 si dispone il FF βr

M_4 si trasferisce all'indirizzatore W il contenuto del registro 0

M_5 si dispone il FF Kz

M_9 il contatore Cp conta + 1. Il decodificatore di p-pulsi fornisce il segnale p_1

p.d.c. p_1 M_0 l'indirizzatore W conta — 1

M_0 si dispone (o ripone) il FF Zc

M_3 si dispone (o ripone) il FF Zr

M_3 si ripongono i FF Up e Ud

M_4 si legge in Memoria e si staticizza il carattere letto nei FF Up (Ud)

M_4 si esegue il primo confronto nel confrontatore Cr

M_7 si ripone il FF IW (conta in avanti) con βw.P_1.ΔSEP

M_7 si rigenera in Memoria il carattere letto al precedente mastro M_4

M_9 il contatore Cp conta + 1
Il decodificatore di p-pulsi fornisce il segnale p_2

p.d.c. p_2 M_0 l'indirizzatore W conta + 1

M_0 si ripone (dispone) il FF Zc

M_3 si ripone (dispone) il FF Zr

M_3 si ripongono i FF Up e Ud

p_2 M_4 si legge in Memoria e si staticizza il carattere letto in Ud (Up)

M_4 si esegue il secondo confronto nel confrontatore Cr

M_7 si rigenera in Memoria il carattere letto al precedente mastro M_4

M_9 il contatore Cp conta + 1. Il decodificatore di p-pulsi fornisce il segnale p_3.

Nei successivi p.d.c. si eseguono le stesse operazioni finchè dal confrontatore Cr non si ottiene il segnale I∅I, al mastro M_4 dell'ultimo p.d.c. della ricerca.

Al mastro M_4 è presente il segnale $S\eta$ (ΔRI.I∅I.βw.$\overline{P_1}$)

Al mastro M_8 si ripone il FF Kz (ΔRI.βw.$\overline{P_0}$.I∅I)

Al mastro M_9 si azzera il contatore Cp

Al mastro M_9 si dispone il FF p_0

Fase β ed η

p.d.c. p_0 M_0 non si azzera l'indirizzatore W perchè presente il segnale $S\eta$

M_0 l'indirizzatore V non conta

M_2 si dispone il FF η. Si ripone il FF IW (conta avanti)

M_2 viene a mancare il segnale di pilotaggio pMp (pMd) in Memoria

M_3 si ripongono i FF Up (Ud)

M_5 si dispone il FF Kz

M_9 si ripone il FF p_0

M_9 il contatore Cp conta + 1. Il decodificatore di p-pulsi dà il segnale p_1

p.d.c. p_1 M_0 l'indirizzatore W conta + 1

M_0 l'indirizzatore V non conta

M_0 è presente il segnale pT_I di consenso al pilotaggio dei registri T_1

p_1 M_4 il contenuto dei FF unità dell'indirizzatore W passa nei FF di uscita da Memoria Ud (Up)

M_7 si scrive dai FF Ud (Up) in registro T_9 il carattere in essi contenuto

M_9 il contatore Cp conta + 1
Il decodificatore di p-pulsi dà il segnale p_2

p.d.c. p_2 M_0 l'indirizzatore W non conta

M_0 l'indirizzatore V conta + 1

M_3 si ripongono i FF Up, Ud

M_4 il contenuto dei FF decine dell'indirizzatore W passa nei FF Up (Ud)

M_7 si scrive nel registro T_9 il contenuto dei FF Up (Ud)

M_9 il contatore Cp conta + 1
Il decodificatore dei p-pulsi fornisce il segnale p_3

p.d.c. p_3 M_0 l'indirizzatore W non conta

M_0 l'indirizzatore V conta + 1

M_3 si ripongono i FF Up, Ud

M_4 il contenuto dei FF centinaia dell'indirizzatore W passa ai FF Ud (Up)

M_7 si scrive nel registro T_9 il contenuto dei FF Ud (Up)

M_9 il contatore Cp conta + 1
Il decodificatore dei p-pulsi fornisce il segnale p_4

p.d.c. p_4 M_0 l'indirizzatore W non conta

M_0 l'indirizzatore V conta + 1

M_3 si ripongono i FF Up, Ud

M_4 il contenuto dei FF migliaia dall'indirizzatore W passa ai FF Up (Ud)

M_7 si scrive nel registro T_9 il contenuto dei FF Up (Ud)

M_7 si scrive il bit gT sull'ultimo carattere introdotto nel registro T_9

p_4 M_7 si dispone il FF φ

M_7 si ripone il FF Kz

M_9 si azzera il contatore Cp

M_9 si dispone il FF p_0

M_9 si ripone il FF βw

M_9 si ripone il FF η

M_9 si dispone il FF α

Disponendosi il FF α si passa alla preparazione dell'istruzione seguente.

È opportuno osservare che se si usano caratteri P, Q, Z, Y nei FF Lu come caratteri condizionatori della ricerca non c'è pericolo che essi vengano interpretati come caratteri condizionatori della lunghezza dell'operando in Memoria, perchè la determinazione della lunghezza per mezzo dei caratteri P, Q, Z, Y esclude le istruzioni di ricerca.

14.5. **Istruzione RIi** (LL IIII T7)

Decodificato ΔSEM; ΔRI

È perfettamente analoga all'istruzione di ricerca avanti; procede però per indirizzi decrescenti di Memoria.
L'unica differenza rispetto alla RIa è che in $\beta w.P_1.M_7$ non si ripone il FF IW.

15. ISTRUZIONI DI SALTO

15.1. Generalità

Le istruzioni di salto alterano la normale sequenza di esecuzione delle istruzioni del programma in corso obbligando l'elaboratore ad abbandonare il programma al punto in cui si trova per riprenderlo in corrispondenza all'indirizzo specificato nell'istruzione di salto.

L'esecuzione del salto può essere incondizionata o subordinata al verificarsi di particolari condizioni.

L'indirizzo dell'istruzione da cui si esegue il salto viene immagazzinato in un registro T specificato dalla stessa istruzione di salto.

15.2. Salto condizionato a confronto

L'istruzione di salto segue una istruzione di confronto. Essendo quattro i possibili risultati di un confronto, si hanno quattro istruzioni di salto condizionato a confronto.

Istruzione SC $>$ (STb IIII Ta0) ($ICV = I_1.I_5.IM >$)

Salta se a seguito del confronto operato nell'istruzione precedente è presente il segnale IM $>$ (FFC$\neq$ disposto e FF C $<$ riposto)

Istruzione SC $<$ (TTb IIII Ta0) ($ICV = I_1.I_6.IM <$)

Salta se a seguito del confronto operato nell'istruzione precedente è presente il segnale IM $>$ (FFC$\neq$ disposto e FF C $<$ disposto)

Istruzione SC $=$ (XTb IIII Ta0) ($I_2.I_5.\overline{C \neq}$)

Salta se a seguito del confronto operato nell'istruzione precedente è rimasto riposto il FF C$\neq$

Istruzione SC $\neq$ (YTb IIII Ta0) ($I_2.I_6.C \neq$)

Salta se a seguito del confronto operato nell'istruzione precedente è rimasto disposto il FF C$\neq$.

15.3. Salto condizionato al valore del contenuto dell'accumulatore

In questo caso l'istruzione di salto segue una istruzione del tipo MA, + MA, — MA, + X, — X.

A seguito di una delle istruzioni indicate il contenuto dell'Accumulatore può essere 0, < 0, > 0.

Le tre indicazioni sono fornite dallo stato dei FF: Za, Sa, Cn.

Lo stato dei FF Sa, Cn dà luogo ai segnali IA + ; IA — indicanti che il numero in Accumulatore è rispettivamente positivo o negativo.

Il FF Za, riposto all'inizio delle istruzioni MA, + MA, —MA, + X, — X si dispone se il numero risultante in Accumulatore è $\neq$ 0. Il contenuto dell'Accumulatore viene preso in esame dalle seguenti istruzioni di salto:

Istruzione SA > 0 (UTb IIII Ta0) ($I_1.I_7.Za.IA+$)

Salta se il contenuto dell'Accumulatore è > 0 (presenza dei segnali Za e IA+)

Istruzione SA < 0 (WTb IIII Ta0) ($I_2.I_{10}.Za.IA-$)

Salta se il contenuto dell'Accumulatore è < 0 (presenza dei segnali Za e IA—)

Istruzione SA = 0 (VTb IIII Ta0) ($I_1.I_8.\overline{Za}$)

Salta se il contenuto dell'Accumulatore è = 0 (presenza del segnale $\overline{Za}$).

15.4. **Salto condizionato a risultato**

In questo caso l'istruzione di salto segue una istruzione del tipo:

MA, MT, IT, MEM, Y

+MA, +MT, +IT, +MM, +AM, +TM

—MA, —MT, —IT, —MM

+X, —X

+LD, XLD, XLN

A seguito di una delle istruzioni indicate si dispone il FF Iz, se il risultato delle operazioni è stato diverso da zero.

Lo stato del FF Iz viene esaminato con due possibili istruzioni di salto:

Istruzione SR = (ΘTb IIII Ta0) ($I_1.I_{11}.\overline{Iz}$)

Salta se il risultato delle operazioni relative ad una delle istruzioni precedentemente indicate è stato zero (presenza del segnale $\overline{Iz}$)

Istruzione SR $\neq$ (. Tb IIII Ta0) ($I_1.I_{10}.Iz$)
Salta se il risultato delle operazioni relative ad una delle istruzioni precedentemente indicate è stato diverso da zero (presenza del segnale Iz).

15.5. Salto condizionato ad « overflow »

L'indicazione « overflow » può aversi a seguito delle seguenti istruzioni che pongono il risultato dell'operazione in Memoria: +MM, —MM, +AM, +TM, o a seguito delle seguenti istruzioni che pongono il risultato in un registro T: +MT, —MT, +IT, —IT.

Nelle istruzioni che pongono il risultato in Memoria la lunghezza del risultato è definita dalla istruzione stessa. L'indicazione « overflow » si ha se in corrispondenza all'ultimo carattere operato si dispone il FF Om; esso può indicare che la lunghezza del risultato supera quella prefissata, o che la lunghezza del risultato è inferiore a quella prefissata.

Nelle istruzioni che pongono il risultato in un registro T la lunghezza del risultato può essere fissata con un numero inferiore a 10 (istruzioni ΔT) o non deve superare il valore 5 (istruzioni ΔI). L'indicazione « overflow », data dal FF OT in corrispondenza all'ultimo carattere operato, indica che la lunghezza del risultato supera od è inferiore rispettivamente al valore prefissato (ΔT) o a 5 (ΔI).

Le istruzioni di salto condizionate allo stato dei due FF Ot, Om sono le seguenti:

Istruzione STO (ZTb IIII Ta0) ($I_2.I_7.\overline{Ot}$)
Salta se non c'è stato overflow in T (FF Ot riposto)

Istruzione SOM ('Tb IIII Ta0) ($I_2.I_8.Om$)
Salta se c'è stato overflow in Memoria (FF Om disposto).

15.6. Salto condizionato ad errore particolare

L'istruzione di salto condizionato ad errore ha luogo se c'è indicazione di errore a seguito di una generica operazione.

Il corrispondente FF di errore viene riposto, dopo che è stato eseguito il salto relativo, al mastro M_5 con il consenso XEC = $I_3.\alpha.P_0$ (cioè in $\alpha.P_0$ dell'istruzione che segue il salto).

Fanno parte di questo gruppo le istruzioni:

Istruzione SEME (J Tb IIII Ta0) ($I_3.I_5.Ee$)

Salta se vi è stato errore di disparità nei caratteri entranti in Memoria quando essa opera con l'esterno (FF Ee disposto). Il FF Ee viene riposto con $XEC.I_5$

Istruzione SEMI (*Tb IIII Ta0) ($I_3.I_{11}.Ei$)

Salta se vi è stato errore di disparità nei caratteri entranti in Memoria quando questa opera con l'interno (FF Ei disposto). Il FF Ei viene riposto con $XEC.I_{11}$

Istruzione SEUM (ε Tb IIII Ta0) ($I_3.I_{10}.Em$)

Salta se vi è stato errore di disparità nei caratteri in uscita da Memoria sul canale interno (FF Em disposto). Il FF Em viene riposto con $XEC.I_{10}$

Istruzione SEA (K Tb IIII Ta0) ($I_3.I_6.Ea$)

Salta se vi è stato errore di disparità nei caratteri in uscita da Accumulatore o registri T (FF Ea disposto). Il FF Ea viene riposto con $XEC.I_6$

Istruzione SEU (L Tb IIII Ta0) ($I_3.I_7.Eu$)

Salta se vi è stato errore in U.A.D. (FF Eu disposto). Il FF Eu viene riposto con $XEC.I_7$.

15.7. Salto condizionato ad errore generale

L'istruzione di salto condizionato ad errore generale ha luogo se si è avuto un qualsiasi errore durante lo svolgimento di una sequenza di programma. Tutti i FF di errore vengono riposti solo se l'istruzione a cui rinvia il salto non è a sua volta un salto.

Fanno parte di questo gruppo le istruzioni:

Istruzione SE (β Tb IIII Ta0) ($I_3.I_9.IE$)

Salta se vi è stato un qualsiasi errore, anche relativo agli organi esterni (presenza del segnale IE). È $IE = Ei + Ee + Em + Ea + Eu + En + El + EGO$; dove non sono ancora noti i segnali: En, errore qualsiasi nei nastri magnetici; El, errore nell'introduzione da fotolettore di consolle; EGO, errore qualsiasi in un governo in linea.

Se l'istruzione seguente non è un salto, si ha il segnale XEE che ripone tutti i FF di errore.

È XEE $= I_9.3Z$ dove è $3Z = \alpha.P_1.I_3.\overline{\Delta Z}.\Delta ZR$

ΔZR è un FF disposto con $\Delta Z.P_0.\beta w.M_9$, quindi solo nelle istruzioni di salto, e riposto con $RN_0.M_9$ ($RN_0 = \alpha.P_1$); questo FF ricorda quindi in $\alpha.P_1$ che l'istruzione precedente è stata un salto; se l'istruzione attuale non è un salto ($\overline{\Delta Z}$) si può quindi avere il segnale 3Z donde XEE.

Istruzione SEGI (+ Tb IIII Ta0) ($I_{12}.I_{15}$.EgI)

Salta se vi è stato un errore qualsiasi durante l'esecuzione di una sequenza di 1° programma (FF EgI disposto). Il FF EgI viene riposto col segnale $\alpha.P_0.I_{12}.I_{15}$ al mastro M_5.

Se l'istruzione seguente non è un salto, si ripongono tutti i FF di errore col segnale:
XEE $= \overline{EgII} . \overline{EgIII} . I_{15} . I_{12}z$ dove è $I_{12}z = \alpha.P_1.I_{12}.\overline{\Delta Z}.\Delta ZR$.

Istruzione SEGII (— Tb IIII Ta0) ($I_{12}.I_{16}$.EgII)

Salta se vi è stato un errore qualsiasi durante l'esecuzione di una sequenza di 2° programma (FF EgII disposto). Il FF EgII viene riposto col segnale $\alpha.P_0.I_{12}.I_{16}$ al mastro M_5.
Se l'istruzione seguente non è un salto, si ripongono tutti i FF di errore col segnale:
XEE $= \overline{EgI} . \overline{EgIII} . I_{16} . I_{12}z$

Istruzione SEGIII (/ Tb IIII Ta0) ($I_{11}.I_{16}$.EgIII)

Salta se vi è stato un errore qualsiasi durante l'esecuzione di una sequenza di 3° programma (FF Eg III disposto). Il FF EgIII viene riposto col segnale $\alpha.P_0.I_{11}.I_{16}$ al mastro M_5.
Se l'istruzione seguente non è un salto, si ripongono tutti i FF di errore col segnale:
XEE $= \overline{EgI} . \overline{EgII} . I_{16}.I_{11}z$, dove è $I_{11}z = \alpha.P_1.I_{11}.\overline{\Delta Z}.\Delta ZR$.

15.8. Salto condizionato da « console »

Le seguenti istruzioni sono condizionate al consenso dato manualmente da « console ».

Istruzione SE 1 (π Tb IIII Ta0) ($I_4.I_9.E_1$)

Istruzione SE 2 ($ Tb IIII Ta0) ($I_4.I_{11}.E_2$)

Istruzione SE 3 (! Tb IIII Ta0) ($I_1.I_{12}.E_3$)

Istruzione SE 4 (& Tb IIII Ta0) ($I_2.I_{12}.E_4$)

15.9. Salto condizionato a segnali provenienti dal GUN

Istruzione SEN (M Tb IIII Ta0) ($I_3.I_8$.XSA; XSA = En.Nf)

Salta se vi è stato un errore su unità a nastro (disposto con il mastro IEN δ da GUN il FF En in U.C. e segnale Nf dal GUN).

Il FF En viene riposto con XEC.I_8.Nf al mastro M_5.

Istruzione SFS (N Tb IIII Ta0) ($I_4.I_{10}$.Fs.Nf)

Salta se è terminata la sequenza su nastro magnetico (si dispone il FF Fs in U.C. all'arrivo del carattere ? da nastro).

Il FF Fs viene riposto all'inizio dell'istruzione successiva, ed inoltre all'inizio della fase esecutiva di qualsiasi istruzione di GUN.

Istruzione SFI (η Tb IIII Ta0) ($I_4.I_{12}$.XSC; XSC = Fi.Nf)

Salta se sono terminate le informazioni su nastro magnetico (si dispone il FF Fi in U.C. all'arrivo del carattere ⊗ dal GUN).

Il FF Fi viene riposto all'inizio dell'istruzione successiva, ed inoltre all'inizio della fase esecutiva di qualsiasi istruzione di GUN.

Istruzione SNL (P Tb IIII Ta0) ($I_4.I_6$.∅NL.Nf)

Salta se è terminato il nastro sul quale si è compiuta l'ultima lettura (segnali ∅NL e Nf del GUN).

Istruzione SNS (Q Tb IIII Ta0) ($I_4.I_7$.XSD; XSD = ∅NR.Nf)

Salta se è terminato il nastro sul quale si è compiuta l'ultima scrittura (segnali ∅NR e Nf del GUN).

Tutte le cinque precedenti istruzioni di salto possono essere eseguite, cioè effettuare o no il corrispondente salto di indirizzi, solo purchè l'eventuale istruzione in corso nel GUN sia finita; ciò perchè durante l'esecuzione di una istruzione di GUN può aversi in qualsiasi momento la segnalazione che crea la condizione di salto (un errore, il fine informazione, ecc.).

Si ha quindi il segnale

$$\mathrm{XEW} = I_3 I_8 + I_4.I_{10} + I_4.I_{12} + I_4.I_6 + I_4.I_7$$

che decodifica tutte e sole le cinque istruzioni precedenti.
Se l'istruzione in corso nel GUN non è finita (segnale $\overline{Nf}$ presente) si dispone con XEW . $\overline{Nf}$. $\beta w.P_0.M_4$ il FF Ni se l'istruzione è stata letta sul 1° programma (ulteriori condizioni di set $\overline{\alpha p}.\overline{\alpha_2}$), il FF Ne se è stata letta sul 2° programma (α_2), il FF N_3 se è stata letta sul 3° programma (αp). È già noto l'effetto di tali FF.

Di conseguenza nel verificatore di salto è sempre stata aggiunta la condizione Nf per ottenere il salto verificato.

Istruzione SNO (n Tb IIII Ta0)

Salta se l'unità a nastro n non è in grado di ricevere informazioni perchè in riavvolgimento o fuori servizio (disposto il FF No in U.C.).

Sull'or del verificatore si ha l'and corrispondente $(\Delta Z+\Delta ZR).\overline{fLd}.$ No.XE β, dove è $\overline{XE\,\beta} = I_9 + I_{12} + I_{11}.I_{13} + I_{11}.I_{14} + I_{11}.I_{16}$. quindi tenendo presente che già vi è $\overline{fLd}$, il segnale XEβ esclude i casi con carattere in P_8 uguale a:

# ÷ α γ	per effetto di I_9
+ — = ,	per effetto di I_{12}
δ	per effetto di $I_{11}.I_{13}$
%	per effetto di $I_{11}.I_{14}$
/	per effetto di $I_{11}.I_{16}$

Si vede così che XEβ.$\overline{fLd}$ individuano tutti i caratteri corrispondenti ad una unità a nastro, ed in più il carattere zero.

Istruzione SCA ((Tb IIII Ta0) $(I_3.I_{12}.ICA)$

Salta se è presente una qualsiasi delle condizioni:

nastro occupato, fine nastro lettura, fine nastro scrittura, fine sequenza, fine informazione, errore qualsiasi, condizione esterna 4, cioè se è presente il segnale:

$$ICA = No + \emptyset NL + \emptyset NR + Fs + Fi + IE + E_4$$

Se l'istruzione seguente non è un salto si ripongono tutti i FF di errore col segnale XEE = $3Z.I_{12}$; inoltre si ripongono i FF No, Fs, Fi sempre con $3Z.I_{12}$.

15.10. Salto incondizionato

Le istruzioni di salto incondizionato hanno esecuzione senza necessità di alcun consenso.

Istruzione SI ( ) Tb IIII Ta0) ($I_1.I_9$)

Salta all'istruzione di cui si specifica l'indirizzo

Istruzione SIN ( ) Tb IIII Ta ∽) ($I_1.I_9$)

Salta avanti nel programma (vedasi per i dettagli il manuale di programmazione)

Istruzione SN (R Tb IIII Ta0) ($I_4.I_8$)

Non salta. Immagazzina ugualmente in Tb l'indirizzo del carattere di P_8 dell'istruzione. Infatti non si ha il segnale ICV, ma ciononostante si dispone ugualmente il FF η con $\beta w.P_0.I\Delta sn.M_2$; $I\Delta sn = I_4.I_8$

Istruzione S_2P (⊗ Tb IIII Ta0) ($I_2.I_9.\overline{2p}$)

Salta dal 1° programma al 2° programma purchè non sia già in corso una sequenza di 2° programma. Se è già in corso un 2° programma (FF 2p disposto) questa istruzione non può essere eseguita e si dispone il FF Ni con $2p.I\Delta 2p$; $I\Delta 2p = I_2.I_9$

Istruzione $S_2\overset{*}{P}$ (δ Tb IIII Ta0) ($I_{11}.I_{13}.\overline{2p}$)

Salta dal 1° al 2° programma purchè non sia già in corso una sequenza di 2° programma. Se è già in corso un 2° programma, l'istruzione $S_2\overset{*}{P}$ non salta e si prosegue nella lettura dell'istruzione che la segue sul 1° programma

Istruzione S_3P o SOL (, Tb IIII Ta0) ($I_{12}.I_{14}.\overline{3p}$)

Salta dal 1° o dal 2° programma al 3° programma purchè non sia già in corso una sequenza di 3° programma. Se è già in corso un 3° programma (FF 3p disposto) questa istruzione non può essere eseguita, e si dispone:

a) il FF Ni, se è stata letta sul 1° programma, con $\Delta SOL.\overline{\alpha_2}.3p$; dove è $\Delta SOL = I_{12}.I_{14}$

b) il FF Ne, se è stata letta sul 2° programma, con $\Delta SOL.\alpha_2.3p$.

Istruzione $S_3\overset{*}{P}$ (= Tb IIII Ta0) ($I_{12}.I_{13}.3p.\overline{N_3}$)

Salta al 3° programma, già in corso, onde riavviarlo. Vedasi in proposito la descrizione del GOL.

15.11. **Salto condizionato a segnali provenienti dal GOL**

Istruzione SEL (γ U IIII Ta0) ($I_9.I_{14}$.EOL)

Salta se vi è errore su unità « on line » U (presenza del segnale EOL) proveniente dal GOL.. Non si può avere trasferimento dell'indirizzo del carattere di P_8 in un registro; si impedisce infatti il disporsi del FF η, essendo presente XQQ = $I_9.I_{14}$

Istruzione SUO (% U IIII Ta0) ($I_{11}.I_{14}.UAO.\overline{ROL}$)

Salta se l'unità « on line » U è occupata (presenza del segnale UAO proveniente dal GOL). Si impedisce anche qui il disporsi del FF η con XQQ = $I_{11}.I_{14}$.

16. ISTRUZIONI DI SALTO DETTAGLIATE

16.1. **Istruzione SC >** (S Tb IIII Ta0)

Decodificati ΔZ, I_1, I_5

Questa istruzione permette di effettuare il salto all'indirizzo IIII modificato dal contenuto in Ta immagazzinando l'indirizzo dell'istruzione di salto relativo al p.d.c. p_7 nel registro Tb. Il tipo di salto è individuato dal carattere operato al p.d.c. p_8 della istruzione ed è abilitato dal segnale ICV, che è presente solo se sono soddisfatte le condizioni di salto.

L'istruzione SC> segue una istruzione di confronto e determina il salto nell'esecuzione del programma se nel precedente confronto si è ottenuto il segnale IM>, indicante che il 1° termine di confronto era > del secondo.

Preparazione ed esecuzione della istruzione SC >

Fase α

p.d.c. p_0 Si azzera l'indirizzatore W, vi si trasferisce il contenuto del registro I, si effettua la conta + 10.

p_1 L'indirizzatore W conta — 1; si estrae da Memoria il carattere di funzione e lo si staticizza nei FF F.

p_2 L'indirizzatore W conta — 1.
Si azzera l'indirizzatore V.

Si legge in Memoria il carattere, lo si estrae e si scrive nell'indirizzatore V l'indirizzo del registro Ta modificatore.

p_3, p_4, p_5, p_6 L'indirizzatore W conta — 1. L'indirizzatore V conta + 1 a partire dal p.d.c. p_4. Si legge in Memoria ad ogni p.d.c. e si scrive nel registro 0 l'indirizzo modificato.

p_7 L'indirizzatore W conta — 1. L'indirizzatore V conta + 1. Si legge in Memoria, se ne estrae un carattere e si trasferisce il carattere letto e modificato da Ta nei FF Lu.

p_8 L'indirizzatore W conta — 1. Si azzera l'indirizzatore V. Si scrive nell'indirizzatore V l'indirizzo in-

dividuato dal carattere operato al p.d.c. p_7. Si legge in Memoria e si scrive il carattere letto nei FF Ld.

Se sono verificate le condizioni di salto è presente il segnale ICV. Si azzera il FF Kz. Si azzera il contatore Cp. Si dispone il FF p_0. Si dispone il FF βw.

Fasi β ed η

p.d.c. p_0 Si dispone il FF η e viene a mancare il pilotaggio in Memoria. Non si azzera l'indirizzatore W e non si trasferisce il contenuto del registro 0 all'indirizzatore W.

Si ripone il FF IW.

p_1 L'indirizzatore W conta + 1.

L'indirizzatore V non conta.

È presente il consenso al pilotaggio dei Registri T.

Si trasferisce il carattere unità dai FF dell'indirizzatore W ai FF Up (Ud). Si scrive il carattere staticizzato in Up (Ud) nel Registro Tb all'indirizzo indicato da V.

p_2, p_3, p_4 { L'indirizzatore W non conta.

L'indirizzatore V conta + 1. Si opera come al p.d.c. p_1 il trasferimento dei caratteri contenuti nell'indirizzatore W al registro Tb.

Col p.d.c. p_4 termina l'istruzione di salto SC>.

Nell'istruzione successiva è presente al p.d.c. p_0 della fase α il segnale ICV che scompare col riporsi dei FF Lu, Ld, al p.d.c. p_1.

Si azzera l'indirizzatore W ma non è possibile trasferirvi il contenuto del registro I, per la presenza del segnale ICV. Si trasferisce invece all'indirizzatore W il contenuto del registro 0, cioè l'indirizzo individuato dall'istruzione di salto. Sempre per la presenza del segnale ICV non si dispone in $\alpha.P_0M_2$ il FF Dw: non si ha perciò la conta + 10.

Se non è presente la condizione (IM>) che rende possibile il salto, non si ha il segnale ICV e perciò non si dispone il FF η; in $\beta w.P_0.\Delta Z.\bar{\eta}.M_7$ si dispone il FF φ; nell'istruzione successiva, per l'assenza del segnale ICV, si trasferisce all'indirizzatore W il contenuto del registro I, e quindi non si effettua il salto.

Se il carattere registrato in P_7 è # (registro Tb inesistente) si ha il segnale $\Delta\eta L_1$ (carattere # nel registro lunghezze unità) che impedisce il disporsi del FF η. Si dispone così immediatamente il FF φ, colla condizione già vista per il caso che il salto non sia verificato. Nella fase α dell'istruzione successiva si ha però ugualmente il segnale ICV e si procede perciò come già visto.

16.2. **Istruzione SIN** ( ) Ta IIII Ta ∽)

Decodificati ΔZ, ΔE, I_1, I_9

Con la istruzione SIN si salta avanti nell'esecuzione del programma di un numero di posizioni pari al numero indicato dall'istruzione ai p.d.c. p_3, p_4, p_5, p_6.

L'istruzione consta di due fasi α ognuna seguita da una fase β.

Nelle operazioni viene impegnato un unico registro T in cui rimane immagazzinato alla fine dell'istruzione l'indirizzo relativo al carattere operato al p.d.c. p_1 della fase α.

Svolgimento delle operazioni relative all'istruzione SIN

1ª Fase α

p.d.c. p_0 Si azzera l'indirizzatore W, vi si trasferisce il contenuto del registro I, si effettua la conta + 10.

p_1 L'indirizzatore W conta — 1.

Si estrae da Memoria il carattere di funzione e lo si staticizza nei FF F. Si dispone il FF λ al mastro M_9.

p_2 L'indirizzatore W conta — 1. Si azzera l'indirizzatore V. Si legge in Memoria il carattere relativo all'indirizzo di Ta, lo si estrae e si scrive nell'indirizzatore V l'indirizzo del registro Ta.

Si trasferisce al registro I il contenuto dell'indirizzatore W. Si dispone il FF λs.

Si ripone il FF Kz con $\alpha.P_2.\Delta Z.\lambda$.

Si dispone il FF p_0. Si azzera il contatore Cp, si ripone il FF α e si dispone il FF βw con $\alpha.P_2.\lambda.\Delta Z$.

Fasi β e η

p.d.c. p_0 L'indirizzatore W non si azzera. Si dispone il FF η con $\beta w.P_0.\lambda.\Delta Z$. Manca il segnale di consenso al pilotaggio in Memoria. È impedito il trasferimento del contenuto del registro 0 all'indirizzatore W. Si ripone il FF Iw.

p_1 L'indirizzatore W conta + 1. L'indirizzatore V non conta. È presente il segnale di consenso al pilotaggio dei registri T. Si trasferisce ai FF Up (Ud) il carattere unità dell'indirizzo contenuto in W. Si scrive il carattere staticizzato nei FF Up (Ud) nel registro Ta, individuato dall'indirizzatore V.

p_2, p_3, p_4 L'indirizzatore W non conta. L'indirizzatore V conta + 1. Si opera come al p.d.c. p_1 il trasferimento dei caratteri decine, centinaia e migliaia del contenuto dell'indirizzatore W al registro Ta.

Al p.d.c. p_4 si dispone il FF φ, termina la fase β e si ripone il FF η. Si azzera il FF Kz, si azzera il contatore Cp, si dispone il FF p_0 e si dispone di nuovo il FF α; si ottiene così nel Ta l'indirizzo effettivo del carattere di P_1.

2ª Fase α

p.d.c. p_0 Si azzera l'indirizzatore W. Si trasferisce all'indirizzatore W il contenuto del registro I. L'indirizzatore W non conta + 10 poichè non si dispone il FF Dw.

Il FF Iw si ripone. (Si era disposto col segnale $\overline{Kz}$) con $\alpha .\Delta Z.\lambda.P_0$.

p_1 L'indirizzatore W conta + 1. Si ripongono i FF F. Si ripongono i FF Up (Ud). È presente il segnale di consenso al pilotaggio in Memoria. Si legge in Me-

moria all'indirizzo specificato da W estraendo di nuovo il carattere di funzione dell'istruzione SIN e trasferendolo ai FF Up (Ud) e quindi ai FF F. Si ripone il FF λ. Si dispone il FF Iw.

p_2 L'indirizzatore W conta — 1. Si azzera l'indirizzatore V. Si legge in Memoria il carattere relativo all'indirizzo di W e si scrive nell'indirizzatore V l'indirizzo del registro Ta modificatore.

p_3, p_4, p_5, p_6, p_7, p_8 — Le operazioni relative a questi p.d.c. sono quelle già descritte nella istruzione SC >. Inoltre in $\alpha.P_3.\Delta Z.\Delta E$ si ripone il FF λs.

Al termine della seconda fase α è stato introdotto nel registro 0 un indirizzo uguale alla somma di quello del carattere di funzione della SIN con la costante operata ai p.d.c. p_3, p_4, p_5, p_6 della seconda fase α. Pur essendo presente il segnale ICV la fase α non è seguita dalla fase η; il disporsi di η è impedito dal decodificato ΔE.

Nella fase α successiva all'istruzione SIN si trasferisce all'indirizzatore W il contenuto del Registro 0 per la presenza del segnale ICV, iniziando la preparazione dell'istruzione individuata dalla precedente SIN.

Si può quindi notare che, cambiando opportunamente il carattere in P_8, l'istruzione SIN può essere condizionata, e si possono avere tutti i salti già precedentemente visti; rimane fermo il tipo di salto: cioè avanti di IIII posizioni di Memoria.

L'unico salto non effettuabile colle particolarità della SIN è il salto su nastro occupato; ciò a causa del FF 00 del GUN che si dispone solo nei salti con carattere zero in P_1.

16.3. **Istruzione S_2p** (⊗ Tb IIII Ta 0)

Decodificati ΔZ, I_2, I_9

La istruzione S_2p è posta sempre sul primo programma e determina l'inizio dell'esecuzione del secondo programma a partire dall'indi-

rizzo IIII modificato dal contenuto del registro Ta. Nel registro Tb viene immagazzinato l'indirizzo relativo al carattere operato al p.d.c. p_7 della fase α dell'istruzione S_2p. Se durante lo svolgimento del secondo programma si susseguono due istruzioni che impegnano contemporaneamente il canale esterno, la seconda delle due istruzioni non viene eseguita e si torna alle istruzioni di primo programma.

Il primo programma continua fino a che non si libera il canale esterno. Se nel primo programma si incontra una istruzione S_2p essa, non potendo essere eseguita, viene riciclata fino all'arrivo del segnale STOP sul secondo programma.

Col segnale STOP sul secondo si passa all'esecuzione dell'istruzione S_2p; si effettua cioè il salto al secondo programma che riprende a partire dall'indirizzo in essa specificato.

La lettura dell'istruzione S_2p può aver luogo anche durante lo svolgimento del programma « on line » se la istruzione SOL, che ha determinato l'inizio del p.o.l., era posta sul primo programma. Il ritorno al primo programma può aversi al seguito delle seguenti condizioni:

a) È stato eseguito un ciclo di p.o.l. che ha attivato tutte le unità « on line ».

b) Nell'esecuzione di un ciclo del p.o.l. sono state lette due istruzioni successive che impegnano il canale esterno contemporaneamente.

   In entrambi i casi il primo programma riprende dall'indirizzo che segue quello relativo all'ultimo carattere dell'istruzione SOL.

   Se tra le istruzioni del primo programma si incontra una istruzione S_2p si esegue il salto al secondo programma.

   Questo continua finchè non si realizzano le condizioni necessarie alla ripresa del programma « on line ».

   Il FF 2p rimane però disposto e quindi se il programma « on line » viene di nuovo temporaneamente interrotto si passa direttamente alla ripresa del secondo programma.

Esaminiamo più da vicino i casi precedentemente considerati.

1) **Istruzione di salto S_2p letta durante lo svolgimento del primo programma**

Fase α

Le operazioni relative alla fase α sono le stesse descritte nella istruzione SC $>$ fino al p.d.c. p_7. Al p.d.c. p_8 si ottiene, oltre alle normali operazioni caratteristiche del p.d.c. p_8 della fase α, il segnale XEK che determina il passaggio ad istruzioni del 2° programma, disponendo nella fase α dell'istruzione successiva il FF α_2 assieme al FF α.

Fasi β ed η

Sono caratterizzate dalle stesse operazioni già descritte nella istruzione SC $>$.

Terminata l'istruzione S_2p, si inizia la preparazione dell'istruzione seguente cioè della prima istruzione del secondo programma:

Al p.d.c. p_0 della fase α si trasferisce all'indirizzatore W, azzerato al mastro M_0, il contenuto del registro 0 anzichè quello del registro I per la presenza del segnale ICV. Si inizia cioè ad operare partendo dall'indirizzo specificato dalla istruzione di salto.

Oltre alle normali operazioni caratteristiche di una generica fase α si ottiene:

p.d.c. p_0 è disposto il FF α_2

p_2 si dispone il FF 2p. Si trasferisce al registro M anzichè I il contenuto dell'indirizzatore W, essendoci α_2.

Al termine di questa istruzione continua l'esecuzione del secondo programma poichè il segnale XEK fornito del FF 2p dispone nuovamente il FF α_2 (ripostosi al p.d.c. p_0 della fase β dell'istruzione precedente) e determina il trasferimento all'indirizzatore W del contenuto del registro M.

2) **Istruzione di salto S_2p letta durante lo svolgimento del secondo programma**

È stato già detto come sia possibile la lettura di una istruzione S_2p posta sul 1° programma, durante lo svolgimento del 2° programma.

Con l'inizio della fase preparatoria di una istruzione del 2° programma si dispongono i FF α ed α_2. Se il canale esterno è occupato e quindi il FF βy è disposto e l'istruzione letta impegna il canale esterno, all'M_4, p_5 della fase α si dispone il FF Ne; in βw, p_0 è impedito il trasferimento all'indirizzatore Y del contenuto del registro 0 dall'assenza del segnale $\overline{\beta y}$. Nello stesso p.d.c. p_0 si ripone il FF α_2 e termina la fase β.

Con la nuova fase α non si dispone più il FF α_2 perchè presente il segnale Ne. All'indirizzatore W viene trasferito il contenuto del registro I anzichè quello del registro M e si passa a preparare l'esecuzione di una istruzione di 1° programma.

Se l'istruzione di primo programma è una S_2p essa verrà eseguita ma non darà luogo al salto all'indirizzo in essa specificato per assenza del segnale ICV che è presente solo quando il FF 2p è riposto.

L'esecuzione dell'istruzione S_2p si svolge nel modo descritto a proposito dell'istruzione SC >.

Essa consta di una fase α in cui si staticizza nel registro 0 l'indirizzo a cui si salta, e della fase β, che determina col p.d.c. p_0 per assenza del segnale ICV.

All'M_4, p_0 della fase β si dispone il FF Ni. Nella successiva fase α non è possibile estrarre da Memoria una nuova istruzione, poichè per assenza del segnale $\overline{Ni}$ non si dispone il FF Dw e quindi è impedita la conta + 10 dell'indirizzatore W. La fase α è caratterizzata dalle seguenti operazioni:

p.d.c. p_0 al mastro M_7 si ripone il FF Iw con $\alpha.\overline{\alpha_2}.\overline{\alpha_p}.Ni$

p_1 al mastro M_0 l'indirizzatore W conta + 1
al mastro M_2 si dispone il FF Iw

p_2 al mastro M_0 l'indirizzatore W conta — 1.

Si estrae così da Memoria ancora il carattere di funzione dell'istruzione S_2p, cioè l'istruzione S_2p ricicla. Il FF Ni si ripone in ogni ciclo all'M_9 del p.d.c. p_3 della fase α e si dispone all'M_4 del p.d.c. p_0 della fase β per mezzo del segnale $2p.I\Delta_2p$.

Se al termine di un ennesimo riciclo il FF βy è riposto, con la nuova fase α si dispone il FF α_2 (per la presenza del segnale XEK.Sb.XFD.XFS) e si torna ad eseguire una istruzione di secondo programma. Al periodo di cifra p_0 della fase α i FF Ni e Ne sono entrambi disposti e quindi è impedita la conta + 10 dell'indirizzatore W, dove è stato trasferito il contenuto del registro M.

La fase α è caratterizzata dalle seguenti operazioni:

p.d.c. p_0 all'M_7 si ripone il FF Iw

p_1 all'M_0 l'indirizzatore W conta + 1;
all'M_2 si dispone il FF Iw

p_2 all'M_0 l'indirizzatore W conta — 1.

In questo modo viene ripetuta l'istruzione di secondo programma che non era stata precedentemente eseguita.

Nella fase α all'M_9 del p.d.c. p_3 si ripone il FF Ne.

Il FF Ni non può più riporsi per assenza del segnale $\overline{\alpha_2}$.

Durante l'esecuzione del secondo programma l'indirizzo dell'istruzione S_2p rimane staticizzato nel registro I.

Se si dovessero leggere sul secondo programma ancora due istruzioni che permettono il passaggio al primo programma, riciclerebbe l'istruzione S_2p.

Col segnale di STOP sul secondo programma si ripone il FF 2p e non si dispone più il FF α_2 assieme al FF α per assenza del segnale XEK.

Si estrae quindi dal registro I l'indirizzo corrispondente al carattere Ta dell'istruzione S_2p.

All'M_7 del p.d.c. p_0 della fase α non si esegue la conta + 10 dell'indirizzatore W poichè il FF Ni si ripone all'M_9 del p.d.c. p_3 della fase α.

Ricicla ancora una volta l'istruzione S_2p, che dà luogo al salto all'indirizzo di secondo programma da essa specificato per la presenza del segnale ICV. Questa volta la fase α della istruzione S_2p è seguita da quella η.

Nella successiva istruzione si dispone col FF α anche il FF α_2 per la presenza del segnale XEK, tramite il decodificato della funzione S_2p che dura fino al p.d.c. p_1 dell'istruzione seguente. Con il segnale $\alpha_2 . p_2$ si dispone il FF 2p che assicura la continuazione dell'esecuzione delle istruzioni di secondo programma.

3) **Istruzione di salto S_2p letta durante lo svolgimento del programma « on line »**

L'inizio di ogni istruzione del p.o.l. è caratterizzata dal disporsi dei FF α ed αp che determinano l'estrazione dal registro H dell'indirizzo relativo all'istruzione in corso. Se il FF α non si dispone per assenza del segnale UD (unità disponibile) o per presenza del segnale N_3 ed è riposto il FF 2p, viene trasferito all'indirizzatore W il contenuto del registro I anzichè quello del registro H e si passa a preparare un'istruzione del primo programma.

Se l'istruzione del primo programma è una S_2p, essa verrà eseguita e determinerà il passaggio allo svolgimento del secondo programma.

Le fasi α e $\beta + \eta$ sono caratterizzate dalle stesse operazioni già descritte nella istruzione SC $>$.

Terminata la istruzione S_2p si inizia la preparazione della prima istruzione del secondo programma. Lo svolgimento del secondo programma prosegue finchè non si ottengono i segnali UD o Sb_3; si riprende allora il p.o.l. col disporsi dei FF α ed αp, mentre il FF α_2 rimane riposto per assenza del segnale XFD.

16.4. Istruzione STOP (O X X X X X # 0)

Decodificati ΔZ, $I_4 . I_5 = \Delta st$

L'istruzione STOP sul primo o sul secondo programma, pone fine allo svolgimento del programma ove è posta, mentre può continuare lo svolgimento degli altri programmi.

L'istruzione STOP posta sul programma « on line » segnala alla unità « on line », individuata al p.d.c. p_7 della fase α che sono terminate le operazioni del p.o.l. che la interessano.

Le operazioni relative alla fase preparatoria e alla fase esecutiva dell'istruzione STOP sono quelle di una generica istruzione di salto con condizione non verificata, cioè si dispone il FF φ in $\beta w.P_0$.

Esamineremo i casi di istruzione STOP sul primo programma, sul secondo programma e sul programma « on line ».

a) **Istruzione STOP sul primo programma**

Lo svolgimento del primo programma può aversi senza che siano stati avviati il secondo programma e il programma « on line », od alternarsi ad essi quando gli stessi vengono momentaneamente interrotti da determinate sequenze di programma.

Nel primo caso l'istruzione STOP arresta il primo programma ed essendo esso, l'unico avviato, hanno termine le operazioni dell'elaboratore. Infatti al p.d.c. p_8 della fase α si genera il segnale Δst e al p.d.c. p_0 della fase β si dispone il FF St e si genera il segnale Ist. La presenza del segnale St impedisce al FF α di disporsi alla fine dell'esecuzione dell'istruzione STOP.

Se invece il FF 2p è disposto, cioè se la lettura della istruzione STOP è stata determinata dalla lettura sul secondo programma di due istruzioni che impegnavano entrambe il canale esterno, hanno termine le istruzioni del primo programma, ma può riprendere l'esecuzione del secondo programma non appena il canale esterno è di nuovo libero. Infatti liberandosi il canale esterno si ripone il FF βy e si dispone il FF Sb. Il segnale Sb permette che si dispongano i FF α ed α_2 e si riprenda quindi lo svolgimento del secondo programma, passando all'esecuzione dell'istruzione che ne aveva determinato l'interruzione.

Durante la fase α in cui è avvenuta la ripresa del secondo programma si ripongono i FF Sb ed Ne. Nelle successive istruzioni la fase α si attiva grazie al segnale $\overline{Ne}$, che permette il disporsi dei FF α ed α_2, assicurando la continuazione dello svolgimento del secondo programma.

Analogo è il caso in cui l'istruzione STOP sul primo programma viene letta mentre il FF 3p è disposto.

In tal caso all'arresto del primo programma può seguire la ripresa del p.o.l. con l'arrivo dei segnali UD, Sb_3 che condizionano l'inizio di un nuovo ciclo del p.o.l.

Con i segnali UD, Sb_3 si dispongono infatti i FF α ed αp.

b) **Istruzione STOP sul secondo programma**

Le fasi α e β dell'istruzione STOP sul secondo programma si svolgono come nel caso a), tuttavia il segnale Δst ottenuto al p.d.c. p_8 della fase α, non dispone il FFSt e quindi non si genera il segnale Ist. Al mastro M_4, del p.d.c. p_0 della fase β si ripone il FF 2p e scompare il segnale XEK. In assenza del segnale XEK, nella fase α dell'istruzione successiva alla istruzione STOP non può più disporsi il FF α_2 e cessa quindi lo svolgimento del secondo programma.

La istruzione STOP sul secondo programma può determinare la ripresa dello svolgimento del primo programma se questo non era stato precedentemente fermato da una istruzione STOP. In caso contrario potrà ancora essere ripreso il p.o.l. se il FF 3p è disposto. Se invece il FF 3p è riposto avranno termine le operazioni relative a tutti i programmi della macchina.

c) **Istruzione STOP sul programma « on line »**

La istruzione STOP posta sul p.o.l. non ne arresta lo svolgimento ma disattiva soltanto l'unità « on line » individuata al p.d.c. p_7 della fase α, senza influenzare il funzionamento dei FF 3p ed αp.

Solo quando tutte le unità « on line » sono state disattivate dalle rispettive istruzioni STOP, si genera il segnale Ro che ripone il FF 3p. Il FF αp riposto in M_9, p_0, βw non viene più attivato per assenza del segnale UD e termina quindi il p.o.l.

17. FLIP - FLOP CONDIZIONATORI ISTRUZIONI DI SALTO

17.1. FF Iz

Il FF Iz si dispone, se il risultato è diverso da zero, in seguito ad una delle seguenti istruzioni:

MA, MT, IT, MEM, Y; +MA, +MT, +IT, +MM, +AM, +TM; —MA, —MT, —IT, —MM; +X, —X; +LD, XLD, XLN

Si ripone all'inizio della fase esecutiva delle istruzioni precedenti ed inoltre nelle AoM e ToM.

Occorre analizzare in un primo tempo il segnale XAL; esso comprende tutti i casi in cui l'uscita dell'U.A.D. non deve influire sul FF Iz, e precisamente tutte le istruzioni non comprese fra le precedenti ed inoltre le +X, —X; +LD, XLD, XLN, AoM, ToM. Non si deve inoltre influire sul FF Iz durante la fase η, di trasferimento dell'indirizzatore W al registro T9, nei casi di fine su carattere chiave.

Si hanno così sul reset del FF Iz i seguenti and:

a) $XAA.P_0.\overline{\Delta Z}$. ($XAA = \beta r.\overline{\eta r}.\overline{XAL}$)

b) $\overline{\lambda}.\rho.P_2$ per le istruzioni +X e —X

c) $\Delta X.XAO$ ($XAO = \beta r.\overline{\eta r}.\Delta T.P_0$) per le +LD, XLD, XLN

d) $\Delta E.\Delta P.XAO$ per la ToM

e) $\beta r.\overline{\eta r}.P_0.\Delta A.\Delta E.\Delta P$ per la AoM

f) $\alpha.P_2.\Delta I.\overline{\Delta BU}$ per le +IT, —IT, IT che non hanno il periodo di cifra $\beta w.P_0$.

Si può quindi disporre il FF Iz se il risultato uscente dall'U.A.D. è diverso da zero, cioè è presente il segnale XAM (or dei 6 bits uscenti da U.A.D.) senz'altro nel caso si abbia il segnale XAA.

Per quanto riguarda le moltiplicazioni si deve notare che le cifre di risultato definitivo, le uniche che devono influire sul FF Iz, sono la prima ottenuta in ogni singola sottofase, e tutte quelle dell'ultima sottofase;
si ha così che col consenso $\beta r.\overline{\eta r}.\Delta AX$ il segnale XAM può disporre il FF Iz nei casi:

a) $P_1.\overline{Sm}$ (la prima sottofase è ottenuta in P_1 se non vi è segno da Memoria)

b) P_2.Sm (la prima cifra è ottenuta in P_2 se vi è segno di Memoria)

c) $\emptyset r.\overline{dn}.\overline{P_1}$ (ultima sottofase).

Per le istruzioni logiche il risultato non è ottenuto nell'U.A.D. bensì all'ingresso di accumulatore; perciò nelle ΔX.ΔT, se vi è qualche bit dei BAi si dispone il FF Iz col consenso $\beta r.\overline{\eta r}.\overline{P_0}.\overline{Ap}$.

17.2. FF Za

Il FF Za si dispone, se il risultato è diverso da zero, in seguito ad una delle seguenti istruzioni con risultato in accumulatore:
MA, —MA, +MA, —X, +X.

Si ripone all'inizio della fase esecutiva delle istruzioni precedenti ed inoltre nella AoM.

Il significato dei comandi di set e di reset è immediato.

17.3. FF Ot

Il FF Ot si dispone se si è avuto overflow nel registro T operato in una delle istruzioni +MT, —MT, +IT, —IT.

Si ripone all'inizio della fase esecutiva di tali istruzioni.

Il significato dei comandi di set e reset è già stato visto a proposito delle singole istruzioni.

17.4. FF Om

Il FF Om si dispone se si è avuto overflow nella Memoria in una delle istruzioni +MM, —MM, +AM, +TM.

Si ripone all'inizio della fase esecutiva di tali istruzioni.

Il significato dei comandi di set e reset è già stato visto a proposito delle singole istruzioni.

17.5. FF Ee

Indica che vi è stato errore sul canale di ingresso esterno a Memoria.

Viene disposto in M_{10} nei casi:

a) $EMd.Zc.PMdr.\overline{XA\emptyset}$ (ingresso dall'esterno alla Memoria dispari)

b) $EMp.\overline{Zc}.PMpr.\overline{XA\emptyset}$ (ingresso dall'esterno alla Memoria pari)

Il segnale $\overline{XA\emptyset}$ è sempre presente, tranne nell'ultimo p.d.c. di una ΔMM in cui è disposto il FF Am. Infatti in tale p.d.c. sul canale esterno esce un carattere che non deve essere operato; si ricorda che Am significa che l'uscita sul canale esterno deve essere operata nel p.d.c. successivo.

Non si segnala un eventuale errore di tale carattere, dato che tale posizione di Memoria è pilotata per esigenze della logica interna e non per richiesta della programmazione.

17.6. **FF Ei**

Indica che vi è stato errore sul canale di ingresso interno a Memoria; le condizioni di set sono quindi analoghe a quelle di Ee; vi è solo uno scambio per Zc e $\overline{Zc}$ ed è eliminato il segnale $\overline{XA\emptyset}$.

17.7. **FF Em**

Indica che vi è stato errore sul canale di uscita interno di Memoria. L'indicazione di errore, condizionata ai pilotaggi di Memoria e a Zc, dispone il FF Em:

a) quando si ha la rigenerazione in Memoria di quanto letto sul canale interno (segnale XMG), escluso il caso della fase esecutiva della NAN in cui l'indirizzatore W è in realtà collegato al canale di uscita esterno

b) nelle operazioni aritmetiche con risultato in Memoria (ΔΣ), in cui cioè non si ha rigenerazione di Memoria, e però l'uscita di Memoria è uno degli operandi.

17.8. **FF Ea**

Indica che vi è stato errore in uscita da accumulatore e da complementatore d'accumulatore.

Viene disposto in M_8 se è presente IEA, si ha il pilotaggio dei registri (PTS) o dell'accumulatore (PAS) ed è consentita l'uscita (CAU + gAu).

Inoltre anche nelle ΔMM con Am, che cioè usano i FF di uscita accumulatore.

17.9. **FF Eu**

Staticizza all'M_2 l'indicazione IEU di errore nell'unità aritmetica. Il segnale IEU è presente:

a) Se si è avuta indicazione di errore nella prova del 3 (IE3) nelle istruzioni aritmetiche (ΔK) escluse le operazioni logiche ($\overline{\Delta TX}$)

b) Se si è avuta indicazione di errore di disparità in Rm (IERm) in $\rho.P_2$

c) Se durante un'operazione aritmetica (ΔK), escluse le operazioni logiche ($\overline{\Delta TX}$) e la IT ($\overline{\Delta IB}$), è uscito dall'unità aritmetica decimale un carattere non numerico; tutte le configurazioni non numeriche sono individuate da $\bar{a}b$ o $a\bar{b}c$. Quest'ultima configurazione è però ammessa quando si ha segno in Memoria; infatti in una +MM (o —MM), se si ha il segno in uno degli operandi, lo si deve avere anche nell'altro; tale segno è introdotto nel Ca ed esce come risultato dalla U.A.D.

La logica della prova del 3 è già stata vista nella « Descrizione logica degli organi » a pag. 125 e seguenti.

Occorre aggiungere alcune considerazioni.

La conta nei vari FF m ed n avviene coi comandi normali di conta avanti ed indietro solo se vi è il consenso $\overline{XED}$.

Non avviene se è presente XED, e precisamente:

a) $\alpha.\overline{P_0}$ Cioè in fase α non si ha il controllo della prova del 3. Si permette però la conta in P_0 per tenere conto dei riporti immagazzinati nell'ultimo p.d.c. nel Kr. Infatti per la logica della prova del 3 si deve tener conto anche dei riporti; questo è il motivo per cui l'eventuale segnalazione di errore si può avere solo in P_1 e non già in P_0

b) aMI.$\overline{\text{bMI}}$.cMI Cioè non si agisce sui contatori in base 3 se è uscito un segno da Memoria

c) ΔI.ISM Cioè non si agisce sui contatori in base 3 se in una ΔI è uscito da Memoria il carattere # o ÷; infatti in tal caso l'operazione è già finita.

Nel comando di conta avanti del contatore connesso all'uscita di U.A.D. sono stati aggiunti i due and:

a) $\Delta I.P_7.ISM.\lambda$ per tenere conto di un eventuale riporto ottenuto in P_6, staticizzato nel FF λ dato che tale riporto viene cancellato in Rr col segnale $\Delta I.ISM$

b) $\Delta I.P_8.ISM.\overline{dMI}.\lambda$ in modo analogo al caso a) per un riporto ottenuto in P_7. Questo è l'unico motivo per cui un riporto ottenuto in P_7 in una ΔI viene staticizzato nel FF λ.

Nei comandi di reset connessi alla logica di conta dei FF m ed n del contatore di Memoria è stato aggiunto il segnale $\overline{XM\alpha}$; infatti quando si ha $XM\alpha$, cioè uscita del carattere chiave da Memoria, si possono avere i comandi di conta indietro o avanti; però, dato che ai fini dell'operazione aritmetica il carattere chiave è uno zero, non si devono avere conte; il reset è stato perciò condizionato a $\overline{XM\alpha}$; il set dei FF non è condizionato a tale segnale essendo già condizionato alla presenza di $\overline{IDM}$ o IDM o ID1M.

L'eventuale segnalazione di errore della prova del 3 è prelevata solo quando la relativa operazione di controllo è terminata, e precisamente nel p.d.c. P_1 della fase successiva; essa può essere η, α, ρ. Si ha quindi il consenso all'uscita della segnalazione di errore: $P_1.\overline{\beta w} + \eta.P_1$.

17.10. FF EgI, EgII, EgIII

Tali FF si dispongono quando avviene un qualsiasi errore durante lo svolgimento rispettivamente del 1°, 2° o 3° programma.
Per poter discriminare a quale programma sia dovuto un particolare errore esistono i seguenti FF:

a) PII e PIII - Si dispongono all'inizio della fase esecutiva interna di una istruzione letta rispettivamente sul 2° o sul 3° programma. Si ripongono in $\alpha.P_1.M_9$ dell'istruzione successiva (oltre che con lo sgancio da console).

   L'assenza di entrambi ($\overline{PII}.\overline{PIII}$) indica che si ha una fase esecutiva interna di una istruzione letta sul 1° programma.

b) YII e YIII - Si dispongono all'inizio della fase esecutiva esterna (cioè col disporsi di βy) di una istruzione letta rispettivamente sul 2° o sul 3° programma.

   Si ripongono col riporsi di βy (oltre che con lo sgancio da console).

L'assenza di entrambi $(\overline{YII.YIII})$ e la presenza di βy indicano che è in corso una fase esecutiva esterna di una istruzione letta sul 1° programma.

c) DTI, DTII, DTIII - Si dispongono all'inizio della fase esecutiva di una istruzione interessante il GUN letta rispettivamente sul 1°, 2° o 3° programma. Si ripongono con la fine esecuzione del GUN (∅emδ) oltre che con lo sgancio da console.

Si hanno ancora i seguenti segnali:

a) H = SEi + SEm + SEa + SEl; cioè l'or dei segnali di set dei FF indicanti un errore dovuto ad una elaborazione sul canale interno (manca soltanto l'errore di unità aritmetica, trattato a parte).

b) X, Y, Z indicanti che è in corso una fase α, β o ρ di una istruzione letta rispettivamente sul 1°, 2° o 3° programma.

Pertanto si dispone EgI nei seguenti casi:

a) Se vi sono i segnali H ed X al M_8; si è cioè avuto un errore di tipo interno durante una fase di tipo interno di una istruzione letta sul 1° programma.

b) Se si ha $\overline{YII}.\overline{YIII}.SEe.M_9$; si è cioè avuto un errore in ingresso di Memoria sul canale esterno durante la fase βy di una istruzione letta sul 1° programma.

c) Se si ha DTI . IENδ; IENδ è un mastro proveniente dal GUN indicante che si è avuto un errore nel GUN.

d) Se si ha $IEU . \overline{PII} . \overline{PIII} . M_1$; cioè se si è avuto un errore di unità aritmetica durante una fase esecutiva di tipo interno di una istruzione letta sul 1° programma; quest'ultimo caso non può essere unito al caso a) dato che la segnalazione di errore nella prova del 3 si ha solo al p.d.c. P_1 della fase successiva a quella in cui si è avuto errore; appunto per tale motivo i FF PII e PIII sono riposti in $\alpha.P_1.M_9$.

I FF EgII ed EgIII si dispongono in maniera perfettamente analoga nei vari casi a, b, c, d; inoltre si dispone EGIII con $EGO.M_9$ (EGO è la segnalazione di errore su unità on-line).

18. ISTRUZIONI DEL TAVOLO DI COMANDO

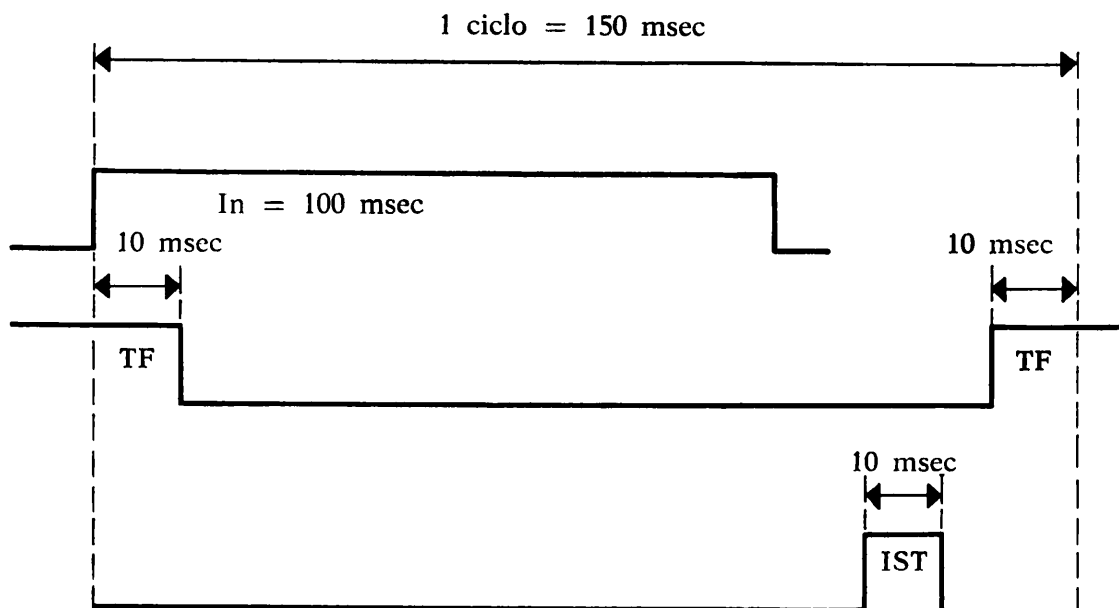
18.1. La telescrivente

La telescrivente è utilizzata come organo di ingresso o di uscita direttamente connesso al tavolo di comando.

Essa può ricevere e stampare 400 caratteri al secondo; il tempo dedicato ad un carattere è quindi di 150 msec, ed è detto « ciclo » di telescrivente. Si deve avere un « ciclo » di telescrivente per ogni carattere stampato, sia per comando dell'U.C. che per comando da tastiera, ed anche per avviare il comando di « ritorno carrello e interlinea ».

Il ciclo di telescrivente corrisponde ad un giro dell'albero principale della macchina, ottenuto collegandolo con il motore tramite un innesto. All'inizio di ogni ciclo deve perciò aversi il comando di « innesto »; tale comando è dato meccanicamente ogni volta che viene premuto un carattere della tastiera, oppure è generato dall'U.C., ogni qual volta occorra, un segnale In che determina, tramite un relé, l'effettuarsi dell'innesto.

Facendo riferimento alla figura si hanno i seguenti segnali scambiati fra telescrivente e calcolatrice:

TF indica che la telescrivente è ferma; scompare circa 10 msec dopo l'inizio di un ciclo, e ricompare circa 10 msec prima della fine del ciclo.

IST È un segnale, detto strobe di telescrivente, della durata di circa 8 ÷ 10 msec che compare verso la fine del ciclo; il fronte di discesa di tale segnale precede di almeno 2 ÷ 3 msec il fronte di risalita del segnale TF. Tale segnale indica che, in seguito alla battuta di un carattere, le barre individuanti il relativo codice sono state posizionate, ed è quindi possibile scrivere il carattere battuto sulla tastiera nella Memoria di U.C.

In È il comando inviato dall'U.C. per avviare un ciclo di telescrivente. Tale comando è assente nei cicli avviati dalla battuta su tastiera di un carattere.

Si hanno inoltre i seguenti segnali scambiati tra U. C. e telescrivente:

IR Indica che si è arrivati al fondo di una riga di telescrivente ed è necessario inviare un comando di « ritorno carrello ed interlinea ». Una riga di telescrivente contiene 72 caratteri; il segnale IR compare dopo il segnale TF di fine ciclo relativo al 71° carattere e prima del segnale IST relativo al 72° carattere; esso è dato da un contatto di fine corsa del carrello.

ISU_1 È il comando di « ritorno carrello ed interlinea ». Dura dall'inizio del relativo ciclo fino al corrispondente IST.

BTL Sono i 6 bits relativi alle barre di codice della telescrivente; rappresentano il carattere che è stato battuto sulla tastiera e che è inviato all'U.C.

It Sono 6 FF che rappresentano il codice di telescrivente del carattere inviato alla stessa dall'U.C.

18.2. **Istruzione CM** (LL IIII T%)

Decodificato ΔAEB

Con la istruzione CM si introducono i caratteri nella Memoria della U.C. tramite telescrivente o fotolettore.

La lunghezza dell'istruzione è definita dai caratteri operati ai p.d.c. p_7, p_8 e staticizzati nei FF Lu, Ld.

Nel caso di lettura da fotolettore la fine dell'istruzione è inoltre determinata dalla lettura del carattere chiave α. L'indirizzo di partenza è quello definito dall'istruzione e modificato dal contenuto del registro T.

Uso del fotolettore

Il fotolettore esegue la lettura della banda perforata con le testine di lettura A e B per cui ogni carattere viene letto due volte. La lettura di un carattere eseguita dalla testina A viene staticizzata nei FF Au dell'U.C. la lettura dello stesso carattere eseguita dalla testina B viene trasferita in Memoria U.C. previo confronto con il contenuto dei FF Au. Se i due caratteri non sono uguali si ottiene il segnale di errore IEl che dispone il FF El.

La distanza fra le due testine di lettura è tale che quando B legge il carattere n, A legge il carattere n + 1.

Alla fine della fase α inizia il moto della banda perforata con il comando $START . FL = \Delta AEB . \beta w . \text{(FL)}$.

Durante il periodo di avvio del moto della banda perforata l'U.C. è ferma al p.d.c. P_0 della fase β; infatti il consenso al riporsi del FF P_0 e alla conta del contatore principale (XLH) manca durante la fase β della CM (ΔAEB).

Il FF Iw è disposto per la conta indietro; infatti la prima conta nell'indirizzatore W deve essere indietro a causa della differenza di una unità in meno negli indirizzi reali rispetto agli indirizzi di programmazione.

L'indirizzatore V è azzerato da $\alpha . P_8 . M_0$. Il FF λ è riposto da $\alpha . P_1 . M_9$.

La linea di ritardo cicla su se stessa.

Quando il 1° carattere arriva sotto la testina di lettura A si hanno i relativi bits ITA; si ottengono pertanto i segnali:

$Icm = ITA . \text{(FL)} . \overline{\lambda}$; $XWT = ICm . \beta w . \overline{Ar}$; $\overline{XW\beta} = \overline{V_7 d} + \overline{V_6 u}$

ed è abilitata la conta dell'indirizzatore V:

$$KV = \Delta AEB . XWT . \overline{XW\beta} . M_1$$

In funzione della conta di V si ottiene:

$V = 0 \div 29$ — si ripongono i FF BTA, BTB (staticizzatori delle letture delle testine A e B) col segnale $XW\alpha = V0d + V1d + V2d$ all'M_9

$V = 30$ — è presente il segnale $\overline{XW\alpha}$ che determina la scrittura del carattere letto nei FF BTA solo nel caso in cui la sua durata superi i 300 μ sec escludendo segnali spuri derivanti da imperfezioni della carta. Infatti se la sua durata è inferiore manca il consenso XWT alla conta dell'indirizzatore V (dato che viene a mancare ICm) e quindi non si può giungere a $V = 30$. Una volta giunti a $V = 30$ la conta dell'indirizzatore V è assicurata da $XWT = \overline{XW\alpha} . \beta w.\overline{Ar}$. La durata del segnale ITA è dell'ordine di 800 μ sec.

$V = 70 \div 79$ — Il contenuto dei FF BTA passa nel transcodificatore Dcm e si ottengono in uscita i BTM

$V = 71$ — Si ripongono i FF Au al M_3

$V = 75$ — Si trasferisce al M_4 il contenuto dei FF BTA transcodificato, cioè i BTM, nei FF Au; si dispone il FF λ al M_9 con $\Delta AEB.\beta w.V7d.V5u$

$V = 76$ — Scompare il segnale $\overline{XW\beta}$. Cessa la conta dell'indirizzatore V.

Si azzera l'indirizzatore V con $\Delta AEB.\beta w.\overline{Ar}.M_7.\overline{ICms}$. ICms è un FF riposto al mastro M_5 con $XW\beta.\overline{ICm}$ e disposto al successivo mastro M_5.

Dopo circa 1200 μsec il 1° carattere viene letto dalla testina di lettura B mentre la testina A legge il 2° carattere. Con la lettura della testina B si hanno i relativi bits ITBb e si ottengono pertanto i segnali:

ICm = ITBb.(FL).λ; XWT; $\overline{XW\beta}$.

L'indirizzatore V può di nuovo contare partendo da 0.

In funzione della conta di V si ottiene:

$V = 0 \div 29$ — Si ripongono i FF BTA, BTB

$V = 30$ — È presente il segnale $\overline{XW\alpha}$ che assicura la conta dell'ind. V e determina la scrittura dei due caratteri

letti dalle testine A e B rispettivamente nei FF BTA e BTB. Disponendosi i FF BTB si ottiene il segnale ITB (or dei BTB) che verrà successivamente utilizzato per riporre il FF P_0, attivare la conta del contatore Cp, dell'indirizzatore W ed abilitare il pilotaggio della Memoria.

V = 60 — Si ripone il FF P_0 al mastro M_9 e contemporaneamente si inizia la conta del contatare Cp. Si ottiene il decodificato P_1.

V = 60 ÷ 69 — Il contenuto dei FF BTB passa nel decodificatore DCm e si ottengono in uscita i BTM.

V = 65 — L'ind. W conta —1 al mastro M_0 col consenso: $\overline{Ar}.\beta w.\Delta AEB.XWV$; $XWV = ITB.V6d.V5u$

Si dispone all'M_0 il FF XMD con $XWV.\overline{Ar}$ e si ottiene il pilotaggio di Memoria PMp (PMd), con $\Delta AEB . XMD . \overline{Ar} . Zc$ ($\overline{Zc}$), durante il successivo V = 66. Si scrivono perciò in Memoria i BTM.

V = 66 — Si effettua il confronto fra i BTM (ottenuti transcodificando i BTB) e i BAu; una eventuale differenza genera il segnale di errore IEl che dispone il FF El col consenso $\Delta AEB.$FL$.\lambda.V6d.V6u.IEl$.

V = 70 ÷ 79 — Il contenuto dei FF BTA passa nel transcodificatore DCm e si ottengono in uscita i BTM.

V = 71 — Si ripongono i FF Au al M_3.

V = 75 — Si trasferisce al M_4 il contenuto dei FF BTA transcodificato, cioè i BTM, nei FF Au.

V = 76 — Scompare il segnale $\overline{XW\beta}$. Cessa la conta dell'indirizzatore V. Si azzera l'ind. V.

Si è così ottenuto il trasferimento da fotolettore a Memoria del 1° carattere letto sulla banda perforata. Per i caratteri successivi si ripetono le stesse operazioni, ma con conta avanti dell'ind. W. Infatti, dopo aver ottenuto con V = 60 il decodificato P_2, al primo M_7 si ripone il FF IW con il consenso $\Delta AEB.\beta w.P_2$.

Se una zona bianca della banda perforata segue sotto le testine di lettura a un gruppo di caratteri letti e precede un altro gruppo di caratteri, alla lettura in B dell'ultimo carattere letto in Memoria, la staticizzazione nei FF Au del carattere ∽ corrispondente a « zona non perforata ».

Alla lettura in A del 1° carattere che segue la zona bianca, si attiva subito la conta dell'ind. V nonostante che il FF λ sia disposto, mediante il segnale:

ICm = ITA.$\overline{\text{ITBb}}$.XTO dove XTO rappresenta il decodificato del carattere ∽ nei FF Au, in and con (FL). Durante tale ciclo di conte nell'ind. V non si ha alcuna lettura dalla testina B; pertanto manca il segnale ITB e non si può avere conta nel Cp, nel W, nè pilotaggio di Memoria.

In questo modo si ottiene la staticizzazione nei FF Au del carattere letto in A, donde la possibilità di ottenere il controllo anche del 1° carattere che segue ad una zona non perforata.

Fine dell'istruzione CM

Come è stato già detto, la fine dell'istruzione può aversi in seguito al segnale emesso dal confrontatore Cl che paragona il contenuto dei FF Lu Ld con la conta del contatore Cp.

Quando l'ultimo carattere viene letto dalla testina B si ottiene con V = 60 al mastro M_0 l'ultima conta di Cp.

Infatti con V = 61 al mastro M_5 si dispone il FF ∅cc.
Le operazioni si susseguono in questo ordine:

V = 65 W conta + 1

V = 66 il carattere staticizzato nei FF BTB viene scritto in Memoria

V = 70 al mastro M_7 si dispone il FF φ. Il segnale φ pone fine all'istruzione in corso e dispone l'U.C. per una nuova istruzione. Il segnale φ arresta il moto della banda perforata riponendo il FF βw e facendo quindi mancare il comando START-FL.

Nel caso in cui si legge su banda perforata il carattere chiave α con la testina B si ha pure l'arresto della istruzione, poichè con V = 69 al mastro M_7 si dispone il FF φ.

Se durante la lettura della banda perforata avviene un errore che dispone in V = 66, M_5 il FF El, ed è presente il segnale da console (AE) (arresto su errore), all'M_7 si dispone il FF φ determinando la fine dell'istruzione e l'arresto del fotolettore.

Uso della tastiera di telescrivente

Se i caratteri vengono introdotti in Memoria con la tastiera della telescrivente non è necessario operare il controllo del carattere scritto, poichè il carattere viene contemporaneamente scritto su carta.

Gli operandi della U.C. interessati dallo scambio di segnali con la telescrivente nell'istruzione CM si trovano inizialmente nel seguente stato:

FF	Ta	disposto
FF	In	riposto
FF	ISU_1	riposto
Univibratori	ISU_2 ISU_3	diseccitati
FF	ISU_4	riposto

Eseguita la battuta del 1° carattere, scompare dopo circa 10 ms. il segnale TF e dopo circa 125 ms. si genera il segnale IST che ripone il FF Ta con XTZ. Contemporaneamente al segnale IST sono presenti nella telescrivente i bits corrispondenti al carattere battuto.

Col segnale $\overline{Ta}$ si ottengono i segnali ICm, ((TAST) . $\overline{Ta}$. $\overline{ISU_2}$), XWT, XWβ e si inizia la conta dell'indirizzatore V. Si genera inoltre il segnale XTF ≡ ITB; XTF = (TAST) . $\overline{ISU_2}$. $\overline{Ta}$.

Quando V = 60 si ripone il FF p_0 e inizia la conta del contatore Cp

V = 65 l'indirizzatore W conta — 1

V = 66 si scrive in Memoria il carattere da Telescrivente; infatti i BTL con (TAST) . ΔAEB passano nel decodificatore e diventano i BTM

V = 76 scompare il segnale $\overline{XW\beta}$ e cessa la conta nell'indirizzatore V.

Dopo circa 135 ms. dall'esecuzione della battuta scompare il segnale IST e la telescrivente cessa di inviare i bits relativi al carattere operato.

Prima della fine del ciclo di stampa, che dura circa 150 ms., compare il segnale TF che dispone il FF Ta; scompare il segnale ICm e si azzera l'indirizzatore V.

Nelle successive battute si opera come per il primo carattere, con conta in avanti dell'indirizzatore W, poichè il FF Iw si è riposto al p.d.c. p_2 della fase β.

Quando viene battuto il carattere corrispondente all'ultima posizione di una riga, la telescrivente deve eseguire l'operazione di ritorno carrello-interlinea. Allo scopo viene generato dalla telescrivente il segnale IR che segue il segnale TF di fine ciclo del 71° carattere e precede il segnale IST del 72° carattere.

Il segnale IR con IST del 72° carattere dispone il FF ISU_4.

All'arrivo del segnale TF alla fine del ciclo corrispondente al 72° carattere, con $TF.\overline{ISU_2}.ISU_4$ si dispone il FF ISU_1 e con $TF.\overline{ISU_2}.ISU_4.\Delta AEB$ si eccita l'univibratore In che determina l'avvio di un ciclo di telescrivente. Con segnale ISU_1 si determina l'inizio del ritorno carrello, e si eccita l'univibratore ISU_2, che rimane eccitato per tutta la durata dell'operazione « ritorno carrello interlinea ». Il FF ISU_1 viene riposto con IST del ciclo di « ritorno carrello interlinea ». Finchè l'univibratore $\overline{ISU_2}$ è eccitato manca il segnale $\overline{ISU_2}$ e quindi è impedito qualsiasi trasferimento da telescrivente a U.C.
Quando compare di nuovo il segnale $\overline{ISU_2}$, può iniziare di nuovo il ciclo di operazioni precedentemente descritto; contemporaneamente viene riposto il FF ISU_4 con $\delta\overline{ISU_2}$.

Con la battuta dell'ultimo carattere alle normali operazioni di trasferimento del carattere da telescrivente a U.C. segue in telescrivente il ritorno carrello poichè, essendo presente il segnale $\emptyset c$, in corrispondenza alla conta 70 dell'indirizzatore V si dispone il FF ISU_1 con $\Delta AEB.(TAST).\emptyset c.\overline{ISU_3}$ e si attiva l'univibratore In con $\Delta AEB.\emptyset c.XTS$ dove è $XTS = (TAST).V7d.\overline{ISU_2}$.

18.3. **Istruzione MS** (LL III T ∅)

Decodificato ΔAUR

Con questa istruzione si scrivono con la telescrivente LL caratteri contenuti in Memoria U.C. partendo dall'indirizzo di Memoria IIII modificato dal contenuto del registro T, individuato ai p.d.c. p_3, p_4, p_5, p_6 della fase α.

Nell'esecuzione di questa istruzione si distinguono per ogni carattere operato una prima fase (τ) durante la quale si effettua il trasferimento di un carattere dalla Memoria ad appositi FF staticizzati ed una seconda fase in cui avviene la stampa del carattere stesso. Le due fasi operative si susseguono fino alla fine dell'istruzione.

Durante la stampa di ogni carattere che ha la durata di circa 150 ms., si può utilizzare l'U.C. per l'esecuzione di altre istruzioni diverse dalla MS. Ciò avviene automaticamente se non è stata disposta in Consolle la chiave « ISOLATO », che impedisce l'attivarsi di una nuova fase α al termine della fase τ relativa ad ogni carattere della istruzione MS. Se è stata inserita la chiave ISOLATO non vengono quindi eseguite altre istruzioni contemporaneamente alla MS.

Preparazione dell'istruzione MS

Fase α

La fase α dura 9 p.d.c. e trova gli organi che provvedono allo scambio dei segnali di temporizzazione tra U. C. e telescrivente nello stato seguente:

FF	Ta	disposto
FF	Tp	disposto
FF	Ts	riposto
FF	In	riposto
FF	ISU_1	riposto
FF	It	riposti
FF	ISU_4	riposto
Univibratore	ISU_2	diseccitato
Univibratore	ISU_3	diseccitato

Operazioni della fase α

p.d.c. p_0 si opera il trasferimento del contenuto del registro I all'indirizzatore W e la conta + 10 dello stesso W

p_1 l'indirizzatore W conta — 1

Si estrae da Memoria il carattere di funzione e lo si staticizza nei FF F, rigenerandolo in Memoria

p_2 l'indirizzatore W conta — 1

Si azzera l'indirizzatore V

Si opera il secondo carattere da Memoria e si individua in V l'indirizzo del registro T modificatore. Si rigenera il carattere operato in Memoria e si dispone il FF λT; tale FF distingue la prima fase τ dalle successive

p_3 l'indirizzatore W conta — 1

Si opera il terzo carattere di Memoria che, modificato dal contenuto del registro T, viene staticizzato nel registro J0. Il carattere letto in Memoria viene rigenerato

p_4 l'indirizzatore W conta — 1

L'indirizzatore V conta + 1

Si opera come al p.d.c. p_3

p_5 si opera come al p.d.c. p_4. Al mastro M_9 si dispone inoltre il FF Ts; tale FF rimane disposto finchè non sono stati stampati tutti i caratteri

p_6 si opera come al p.d.c. p_4

p_7 l'indirizzatore W conta — 1

L'indirizzatore V conta + 1

Si legge il carattere in Memoria e lo si trasferisce, modificato dal contenuto di T, al registro S. Si ha infatti $AJS = \alpha.\lambda T.\Delta AUR.\ (P_7 + P_8)$; CJZ è presente in P_7 per cui si ha la lettura del registro S onde cancellarlo; si seleziona l'ingresso alla decade delle unità dei registri J con

$\lambda T.P_7.\Delta AUR$. Il carattere letto in Memoria viene rigenerato

p_8 l'indirizzatore W conta — 1

Si opera l'ultimo carattere della istruzione estraendolo dalla Memoria e trasferendolo al registro S. Il carattere letto in Memoria viene rigenerato.

Alla fase α segue la fase β che termina al p.d.c. p_0.

Esecuzione dell'istruzione MS

Fase β

p_0 si azzera l'indirizzatore W e vi si trasferisce il contenuto del registro 0. Al mastro M_7 si dispone il FF φ; al mastro M_9 non si dispone il FF α per la presenza dei segnali Ts e Tp; si dispone il FF τ con $Ts.Tp.\varphi.M_9$.

1ª Fase τ

p.d.c. p_0 M_5 si dispone il FF Kz

M_9 si ripone il FF p_0, il contatore Cp conta + 1

p_1 M_0 l'indirizzatore W conta — 1

M_0 si dispone il FF Zc

M_3 si dispone il FF Zr

M_4 è presente il pilotaggio di Memoria PMp (PMd) $= \tau.P_1.\overline{\varphi t}.Zc\ (\overline{Zc})$; si legge in Memoria e si trasferisce il carattere letto ai FF Up (Ud)

M_4 è presente $AJN = \tau.P_1$; perciò si cancella il registro N interrogandolo

M_7 si trasferisce il contenuto di W nel registro N; è presente infatti il segnale $CWJ = \tau.P_1$

M_7 si rigenera in Memoria il carattere letto all'M_4, si ha infatti $XMG = \tau.P_1R$

Si trasferisce il contenuto dei FF Up (Ud) ai FF It attraverso il decodificatore Dmt col consenso $\tau.P_1$

p_2 M_0 l'indirizzatore W non conta, ma viene azzerato con $\tau.M_0.XWX$; $XWX = P_2.\overline{Ar}$

M_4 si trasferisce il contenuto del registro S all'indirizzatore W; si ha infatti $AJS = \tau.P_2$; $CJW = \tau.P_2$

M_7 si ripone il FF Tp con $RTp = \tau.P_2.\overline{RFR}$; è $RFR = RTs.ISU_3$

Si eccita l'univibratore In con RTp

M_0 l'indirizzatore W conta — 1; Iw è su IWI dall'inizio della fase τ

M_7 si trasferisce il contenuto dell'indirizzatore W al registro S; si ha infatti $AJS = \tau.P_3$; $CWJ = \tau.P_3$

M_7 si dispone il FF φ

M_9 si ripone il FF λT ed il FF τ

M_9 se non è stata disposta in console la chiave « ISOLATO » manca il segnale $\overline{ISO}$ ed è quindi presente il consenso $\overline{XFZ}$; si dispone perciò il FF α.

Alla fine di questa prima fase τ si ha nel registro N l'indirizzo effettivo del 1° carattere da stampare; tale carattere è già stato inviato alla telescrivente; nel registro S vi è il numero di caratteri ancora da stampare, cioè la lunghezza specificata nell'istruzione diminuita di una unità.

Mentre l'U.C. può passare a preparare e svolgere altre istruzioni, si iniziano le operazioni per la stampa del carattere staticizzato nei FF It. Dopo circa 125 msec dal momento in cui si è eccitato l'univibratore In, che comanda l'innesto dell'albero della telescrivente, è terminata la stampa del 1° carattere e si ottiene il segnale IST che genera il mastro XTZ; con XTZ si ripongono i FF It e Ta.

Prima della fine del ciclo di stampa si genera il segnale TF che dispone nuovamente il FF Ta e quindi, con Taδ, il FF Tp.

La presenza dei segnali Ts e Tp impedisce che, al termine della fase esecutiva di una ennesima istruzione svolta dall'U.C. durante il ciclo di stampa, si disponga nuovamente il FF α; è presente infatti

il segnale $XFZ = Ts.Tp$ che inibisce il disporsi del FF α. Si dispone invece il FF τ con $Ts.Tp.\varphi.M_9$.

Se è stata disposta in console la chiave « ISOLATO » durante il ciclo di stampa l'U.C. rimane inattiva. Alla fine del ciclo di stampa, essendo presente il segnale $\overline{XFM}$ (che indica assenza di fasi), si dispone il FF τ con $Ts.Tp.\overline{XFM}.$ (ISO) .

2ª Fase τ

p.d.c. p_0 M_0 si azzera l'indirizzatore W con $\tau.P_0.M_0.\overline{\lambda T}$; da notare che questo azzeramento non avviene nella prima fase τ per la presenza di $\overline{\lambda T}$

p_0 M_4 si trasferisce il contenuto del registro N all'indirizzatore W; è $AJN = \tau.P_0.\overline{\lambda T}$; $CJW = \tau.P_0\overline{\lambda T}$

M_7 si ripone il FF Iw su IWA con $\tau.P_0.\overline{\lambda T}$

p_1 M_0 l'indirizzatore W conta + 1

M_4 si legge in Memoria U.C. e si staticizza il carattere letto nei FF Ud (Up)

M_7 si staticizza nei FF It il contenuto dei FF Ud (Up)

M_7 si trasferisce il contenuto dell'indirizzatore W al registro N

p_2 M_0 si azzera l'indirizzatore W

M_2 si dispone il FF Iw su IWI con$\tau.P_2$

M_4 si trasferisce all'indirizzatore W il contenuto del registro S

M_7 si ripone il FF Tp

M_7 si eccita l'univibratore In

p_3 M_0 l'indirizzatore W conta — 1

M_7 si trasferisce il contenuto dell'indirizzatore W al registro S

M_7 si dispone il FF φ

M_9 si ripone il FF τ

M_9 si dispone il FF α.

Col disporsi del FF In al p.d.c. p_2 della fase τ, hanno inizio le operazioni per la stampa del carattere contenuto nei FF It nel modo descritto per il primo carattere.

Quando il carattere da stampare è l'ultimo della riga, la telescrivente deve eseguire l'operazione ritorno carrello-interlinea. In questo caso, dopo il segnale TF di fine ciclo del 71° carattere e prima del segnale IST del 72° carattere, viene generato dalla telescrivente il segnale IR. Tale segnale dispone con IST del 72° carattere il FF ISU_4. La presenza del segnale IR e di ISU_4 impedisce al FF Ta, e di conseguenza al FF Tp, di disporsi all'arrivo del segnale TF di fine ciclo del 72° carattere.

La presenza dei segnali $TF . ISU_4 . \overline{ISU_2}$ dispone il FF ISU_1; con l'and $TF . ISU_4 . \overline{ISU_2} . Ts$ si eccita l'univibratore In; inizia perciò l'operazione di ritorno carrello-interlinea.

Per evitare che si disponga il FF Ta e quindi abbia inizio una nuova fase τ, prima che sia terminata l'operazione di ritorno carrello, si è introdotto l'univibratore ISU_2 che viene eccitato dal segnale ISU_1. Col riporsi di ISU_2 si ripone ISU_4 ed è quindi possibile disporre Ta. Con $Ta\delta$ si dispone Tp e coi due segnali Ts e Tp si dispone nuovamente il FF τ come è già stato visto in precedenza.

Quando si opera l'ultimo carattere si genera nell'indirizzatore W il segnale $I = W$ (cioè zero sia in Wu che in Wd) al mastro M_0 del p.d.c. p_3 della fase τ; infatti nel p.d.c. p_3 della fase τ l'indirizzatore W contiene il numero di caratteri ancora da stampare. Al mastro M_7 dello stesso p.d.c. si trasferisce il contenuto dell'indirizzatore W al registro S; il valore trasferito è zero.

Al mastro M_9 si dispone il FF φt con $\tau . P_3 . I = W$.

Eseguita la stampa dell'ultimo carattere si attiva, per la presenza di Ts e Tp, ancora una volta il FF τ, ma non si estrae da Memoria alcun carattere per la presenza del segnale φt che toglie il consenso al pilotaggio in Memoria.

Al mastro M_7 del p.d.c. p_2, essendo stato trasferito nell'indirizzatore W il registro S che era azzerato, si ripone il FF Ts con $RTs =$
$= \tau . I = W . \overline{\lambda T} . P_2$.

Con lo stesso segnale RTs in and con $\overline{ISU_3}$ si dispone il FF ISU_1 e si eccita l'univibratore ISU_2; col segnale RTp di reset del FF Tp si eccita l'univibratore In; col segnale RTs si ripongono i FF It. Si ottiene perciò l'operazione di ritorno carrello-interlinea.
Con $\overline{Ts}$ si ripone il FF φt.

Col disporsi del FF φ in $\tau.P_3$, essendo Ts riposto, si dispone il FF α anche se è presente il segnale $\overline{ISOL}$. L'U. C. riprende quindi l'esecuzione delle istruzioni di programma.

Le operazioni di telescrivente terminano solo quando l'univibratore ISU_2 è di nuovo diseccitato. Con la presenza del segnale $\overline{ISU_2}$ si dispone il FF Ta e quindi il FF Tp.

Gli organi di U.C. caratteristici della istruzione MS sono perciò ritornati nelle condizioni che inizialmente si erano supposte.

Nel caso si legga nel programma una MS o una CM da console mentre è ancora in corso una precedente istruzione di telescrivente cioè si hanno i segnali Ts o ISU_2) è ovvio che tale istruzione non può essere eseguita. In $\alpha.P_5.M_4$ si dispone in tal caso il FF Ni, Ne o N_3 se l'istruzione è stata letta rispettivamente sul 1°, 2° o 3° programma.
Si ha infatti il segnale $XF\lambda = \Delta AUR + TAST . \Delta AEB$; tale segnale in and con Ts o con $ISU_2 . \overline{ISOL}$ dispone appunto tali FF.

PARTE SECONDA

ISTRUZIONI DI G. U. N.

19. LETTURA E REGISTRAZIONE

19.1. Istruzione LNa (X n IIII T ✻)

Decodificati ΔVER, ΔN

Con questa istruzione si legge sul nastro magnetico n individuato al p.d.c. p_7 della fase α un blocco di caratteri procedendo in avanti e lo si trasferisce in Memoria U. C. partendo dall'indirizzo della istruzione: IIII modificato dal contenuto del registro T.

Lo svolgimento di questa istruzione impegna il solo canale esterno della U.C., per cui contemporaneamente ad essa possono essere eseguite altre istruzioni che impegnino i soli organi interni della U. C.

Preparazione dell'istruzione

Fase α

Durante la fase α si eseguono operazioni che preparano gli organi interni della U.C. e del GUN per l'esecuzione dell'istruzione. Le principali operazioni nella U.C. e nel GUN sono le seguenti:

Unità centrale:

p.d.c. p_0 si azzera l'indirizzatore W e vi si trasferisce il contenuto del registro I. Si opera poi la conta + 10

p_1 l'indirizzatore W conta — 1. Si estrae da Memoria il carattere di funzione e lo si staticizza nei FF F

p_2 l'indirizzatore W conta — 1. L'indirizzatore V viene azzerato. Si estrae da Memoria il carattere del registro T modificatore; se ne individua l'indirizzo scrivendolo nell'indirizzatore V

p_3 l'indirizzatore W conta — 1. L'indirizzatore V non conta. Si leggono in Memoria e nel registro T, selezionato al p.d.c. p_2, i caratteri individuati dai due indirizzatori, W e V. Essi vengono estratti ed operati nella U.A.D. Il risultato è scritto nel registro 0

p_4 l'indirizzatore W conta — 1. L'indirizzatore V conta + 1. Si procede poi come al p.d.c. p_3

p_5 si opera come al p.d.c. p_4

p_6 si opera come al p.d.c. p_4

p_7 l'indirizzatore W conta — 1. L'indirizzatore V conta + 1. I caratteri letti in Memoria e nel registro T vengono estratti ed operati nella UAD.

Il risultato è un carattere che individua l'U.N. in lettura. Esso viene trasferito ai FF Lu della U.C. e inviato al decodificatore U.N. del GUN

p_8 l'indirizzatore W conta — 1. L'indirizzatore V viene azzerato. Si legge in Memoria il carattere corrispondente all'indirizzo dell'indirizzatore W e lo si trasferisce ai FF Ld.

Si ripongono poi il FF Kz ed il FF α, si dispongono il FF βw ed il FF p_0, si azzera il contatore Cp.

G.U.N.

p.d.c. p_0 si dispone il FF Nf

p_1 si genera il segnale Qα che ripone i FF NL, NR

p_2 si dispongono i FF L e ΔN; il segnale L è inviato all'U.C. dove assume il nome δL

p_7 si individua nel decodificatore UN l'UN interessata e si genera il segnale SN_k relativo ad essa

p_8 si genera il segnale SNL e si dispone il FF NL_k che seleziona le testine di lettura della unità nastro interessata. Si interroga il FF No: solo se il FF No non si dispone può aver luogo l'esecuzione dell'istruzione LNa. Se invece il FF No si dispone, cioè l'UN selezionata è impegnata in un riavvolgimento, perviene all'U.C. il segnale No. Nel caso in cui la istruzione è posta sul primo programma si attiva con No in βw, p_0 il FF Ni (U.C.). Al mastro M_7 non si dispone il FF βy e non ha luogo l'esecuzione della istruzione, mentre nella fase α successiva la LNa ricicla, poichè col segnale Ni è impedita la conta + 10 dell'indirizzatore W. Se invece l'istruzione LNa è posta sul secondo programma o sul programma « on line »

col segnale No si dispongono in βw, p_0 rispettivamente i FF Ne, N_3. Essi impediscono la esecuzione dell'istruzione e determinano il passaggio al primo programma nella fase α successiva alla fase β dell'istruzione LNa.

Le operazioni di fase β che impegnano il canale interno della U.C. terminano in βw, p_0. L'U. C. può iniziare la preparazione di una nuova istruzione mentre il canale esterno rimane impegnato dal trasferimento dei caratteri da nastro a Memoria U.C.

Esecuzione dell'istruzione

Operazioni in U.C. e nel GUN

p.d.c. p_0 si azzerano gli indirizzatori W ed Y e vi si trasferisce il contenuto del registro 0. Si dispone quindi il FF βy.

Si genera nel GUN il segnale $Q\beta$. Si dispone in U.C. il FF φ che pone termine alle operazioni che impegnano l'indirizzatore W. Col segnale $Q\beta$ si dispone il FF $V\lambda$ (GUN) e ha inizio il moto in avanti del nastro in lettura.

Si dispone al M_9 il FF βN con $S\beta y.\Delta N$; inoltre si ripone su IYI il FF Iy con $\beta y.\overline{Ce}.M_5$.

Nell'intervallo di tempo fra il segnale $Q\beta$ e la scrittura del primo bit di sincronismo l'U.C. può eseguire istruzioni che impegnano il canale interno con temporizzazione definita dalla propria linea di ritardo chiusa su se stessa. All'arrivo del segnale « Trig N », generato col primo ciclo della linea di ritardo λ del GUN, la linea di ritardo della U.C. viene aperta e sincronizzata su quella λ. Da questo istante le operazioni relative al canale interno ed esterno della U.C. vengono eseguite col sincronismo λ.

La linea di ritardo della U.C. parte al mastro λm della linea λ al suo secondo ciclo. ($M_0 = \lambda m$); infatti al primo ciclo il primo Trig $N \equiv \lambda m$ apre la linea di ritardo riponendo il FF Ie; il secondo Trig N, trovando già presente $\overline{Ie}$ dispone il FF Cl facendo partire la linea di ritardo di U.C.

Dopo la lettura su nastro dei 3 bits di sincronismo si legge su nastro il carattere NP (∽).

Al mastro $\lambda 0$ (M_4) si riempiono i FF F_2 λ (GUN)
$\lambda 3$ (M_7) si dispone il FF NP λ, si ottiene il segnale PrN (GUN)

$\lambda 3$ (M_7) si dispone il FF IB λ (GUN)
M_8 si dispone il FF Ce con $\beta y.\overline{Ce}.\overline{\delta NAN}.PrN$ (U.C.)
M_8 si dispone il FF IB con $\beta y.\overline{Ce}.\overline{\mu}.\overline{\delta NAN}.\overline{\delta PRC}.PrN$ (U.C.)

Segue poi la lettura sul nastro del carattere α.

Al mastro λm (M_0) l'indirizzatore Y conta — 1 con $Ce.\overline{XW\lambda}.M_0$ (U.C.)
M_2 si dispone il FF Iy su IYA con $PrN.\overline{XW\Theta}$ dove è $XW\Theta = J.\delta L$ (U.C.)

$\lambda 0$ (M_4) il carattere α è disponibile nei FF $F_2\lambda$ (GUN)
$\lambda 3$ (M_7) il carattere α viene scritto in Memoria U.C. (il pilotaggio di Memoria è presente con i segnali Ce, βy)

M_{10} si ripone il FF IB con $XN\rho.\overline{\delta NAN}$; dove è $XN\rho = XN\varepsilon$ (α in arrivo dal GUN) + $XN\eta$ (? in arrivo dal GUN) (U.C.)
$\lambda 8$ si ripone il FF IB λ (GUN)

La lettura dei caratteri sul nastro e il loro trasferimento alla Memoria della U.C. continua con conta avanti dell'indirizzatore Y fino all'arrivo del carattere α di fine blocco.

Col carattere α di fine blocco avvengono le seguenti operazioni:

Al mastro $\lambda 0$ (M_4) il carattere α è disponibile nei FF $F_2\lambda$ (GUN)
$\lambda 3$ (M_7) il carattere α viene scritto nella Memoria U.C.
$\lambda 6$ si dispone il FF $\emptyset B\lambda$
M_{10} ($\lambda 6$) si ripone il FF βy con $\overline{\delta NAN}.XF\eta$, dove è $XF\eta = XN\rho.\overline{IB}$; cessa il pilotaggio della Memoria U.C. (U.C.)

M_{10} (λ6) si ripone il FF Ce con $\overline{\beta y \delta}$ (U.C.)
M_{10} (λ6) si ripone il FF βN con $\overline{\beta y \delta}$ (U.C.)
M_{10} (λ6) si avvia la linea di ritardo di U.C. con
$\overline{\beta y \delta}.\overline{\Delta MM}.$(CONT) (U.C.)

Al successivo M_2, essendovi $\overline{\beta y}$, si dispone il FF Ie e la linea di ritardo di U.C. rimane chiusa su se stessa.

Nel GUN segue il carattere NP (ᔕ) di fine blocco.

Al mastro	λ6	si ripone il FF NPλ	(GUN)
	λ6	si attiva l'univibratore ∅λra che, tornando allo stato di riposo, ripone il FF Vλ e toglie il pilotaggio all'U.N. in lettura	
	λ10	si ripone il FF ∅ B λ	(GUN)

19.2. **Istruzione LNi** (Xn IIII T K)

Decodificato ΔVEP

Questa istruzione è perfettamente analoga alla LNa, con moto indietro del nastro magnetico e procedendo in Memoria di U.C. con indirizzi decrescenti.

Le uniche differenze rispetto alla LNa sono le seguenti:
Fase α, p.d.c. p_2 Si dispone il FF J (GUN).

Esecuzione dell'istruzione:

Col segnale Qβ si dispone il FF Vλ (GUN) e, per la presenza del segnale J, ha inizio il moto indietro del nastro in lettura. Alla lettura su nastro del carattere α al M_2 non si dispone su IYA il FF Iy per la presenza di J e di conseguenza l'assenza di $\overline{XW\Theta}$. Il FF Iy rimane riposto su IYI e quindi l'indirizzatore Y procede per conte indietro.

19.3. **Istruzione RN** (X n IIII TM)

Decodificato ΔVXR

Con questa istruzione si trasferisce da Memoria U.C. al nastro magnetico n un blocco di caratteri.

Il trasferimento ha inizio a partire dall'indirizzo IIII modificato dal contenuto del registro T, l'unità nastro interessata alla registrazione viene individuata al p.d.c. p_7 della fase α. Nella registrazione il nastro si muove sempre in avanti per consentire la verifica dei caratteri registrati.

Preparazione dell'istruzione

Fase α

Durante la fase α vengono predisposti gli organi della U.C. e del GUN interessati nell'esecuzione dell'istruzione RN.

Unità centrale

p.d.c. p_0 si azzera l'indirizzatore W e vi si trasferisce il contenuto del registro I.
Si opera poi la conta + 10

p_1 l'indirizzatore W conta — 1. Si estrae da Memoria il carattere di funzione e lo si staticizza nei FF F

p_2 l'indirizzatore W conta — 1. L'indirizzatore V viene azzerato. Si estrae da Memoria il carattere del registro T modificatore; se ne individua l'indirizzo scrivendolo nell'indirizzatore V

p_3 l'indirizzatore W conta — 1. L'indirizzatore V non conta. Si leggono in Memoria e nel registro T selezionato al p.d.c. precedente i caratteri individuati dagli indirizzatori W e V. Essi vengono estratti e operati nella U.A.D. Il risultato viene scritto nel registro 0

p_4 l'indirizzatore W conta — 1. L'indirizzatore V conta + 1. Si procede poi come al p.d.c. p_3

p_5, p_6 si opera come al p.d.c. p_4

p_7 l'indirizzatore W conta — 1. L'indirizzatore V conta + 1. I caratteri letti in Memoria e nel registro T vengono estratti ed operati nella U.A.D. Il risultato

è un carattere che individua la U.N. in registrazione. Esso viene inviato al decodificatore U.N. del GUN

p_8 l'indirizzatore W conta — 1. L'indirizzatore V viene azzerato. Il carattere letto in Memoria viene trasferito ai FF I.d ma non interessa lo svolgimento della istruzione RN.

Si ripongono poi i FF Kz ed α, si dispongono i FF βw e p_0, si azzera il contatore Cp.

G.U.N.

p.d.c. p_0 si dispone il FF Nf

p_1 si genera il segnale Qα che ripone il FF Vρ e i FF NR

p_2 si dispongono i FF ΔN ed R

p_7 si individua nel decodificatore U.N. l'U.N. interessata e si genera il segnale SN_k relativo ad essa

p_8 si genera il segnale SNR e si dispone quindi il FF NR_k che seleziona la testina di registrazione dell'U.N. interessata.

Si interroga il FF No. Se il FF No non si dispone l'istruzione può proseguire. Al p.d.c. p_0 della fase β hanno termine le operazioni che impegnano il canale interno della U.C. e quindi può iniziare una nuova fase α relativa alla istruzione che segue la RN nel programma.

Se il FF No del GUN si dispone, cioè se l'U.N. selezionata è impegnata in un riavvolgimento, perviene alla U.C. il segnale No.

Nel caso in cui l'istruzione RN è posta sul 1° programma si attiva con No in βw, p_0 il FF Ni. Al mastro M_7 non si dispone il FF βy e non ha luogo l'esecuzione della istruzione mentre nella fase α successiva ricicla la istruzione RN, poichè col segnale Ni è impedita la conta + 10 dell'indirizzatore W.

Se invece l'istruzione RN è posta sul 2° programma o sul programma « on line », col segnale No si dispongono in βw, p_0 rispettivamente i FF Ne, N_3.

Essi impediscono l'esecuzione dell'istruzione e determinano il passaggio al 1° programma nella fase α successiva alla fase β della istruzione RN.

Esecuzione della istruzione

Fase β

Operazioni nella Unità centrale e nel GUN

p.d.c. p_0 si azzerano gli indirizzatori W ed Y e vi si trasferisce il contenuto del registro 0.
Si dispone quindi il FF βy.

Si ottiene nel GUN il segnale Qβ essendosi riposto il FF Nf. Si dispone nella U.C. il FF φ che pone termine alle operazioni che impegnano l'indirizzatore W. Col segnale Qβ si succedono nel GUN le seguenti operazioni:

— si dispone il FF Vρ e si ottiene il segnale NA_k che dà inizio al moto del nastro avanti.
Nell'intervallo di tempo che intercorre fra il mastro Qβ e l'invio in U.C. da parte del GUN del 1° mastro di sincronizzazione, l'U.C. svolge le sole informazioni relative al canale interno.

L'esecuzione della istruzione RN inizia quando diseccitandosi l'univibratore Uρ del GUN, inizia il 1° ciclo della linea di ritardo ρ. Il segnale TrN viene inviato alla U.C. col mastro ρm, la linea di ritardo della U. C. si apre e ad ogni mastro ρm successivo cicla sincronizzata con quella ρ del GUN. Il mastro ρm coincide quindi col mastro M_0.

Il GUN provvede a scrivere su nastro i 3 bits di sincronizzazione che precedono il carattere NP (∽).

Nel p.d.c. successivo alla scrittura dell'ultimo bit di sincronismo si susseguono le seguenti operazioni logiche:

Al mastro	$\rho 1$ (M_5)	si dispone il FF NPi	(GUN)
	$\rho 1$ (M_5)	si dispone il FF NPρ; si ottiene il segnale PrN	(GUN)

$\rho 1$ (M_5) si dispone il FF IBρ (GUN)

$\rho 4$ (M_8) si dispone il FF Ce con PrN . βy . $\overline{Ce}$. $\overline{\delta NAN}$ (U.C.)

$\rho 4$ (M_8) si dispone il FF IB con PrN . $\overline{\delta NAN}$. $\overline{\delta PRC}$. $\overline{Ce}$. $\overline{\mu}$. βy (U.C.)

$\rho 7$ si dispongono i FF FR (GUN), scrivendo su nastro il carattere NP

$\rho 8$ si ripone il FF NPi (GUN)

Si registra poi su nastro il carattere α.

Al mastro ρm (M_0) l'indirizzatore Y conta — 1 (U.C.)

ρm_2 (M_2) si dispone il FF Iy come nella LNa (U.C.)

$\rho 0$ (M_4) il carattere α letto in Memoria U.C. è disponibile nei FF Up (Ud)

$\rho 5$ (M_9) si ripone il FF IB con XNδ.$\overline{Ap}$; dove è XNδ = XNΘ + XNλ (rispettivamente α e ? in uscita sul canale esterno di Memoria (U.C.)

$\rho 7$ si dispongono i FF FR per la scrittura su nastro (GUN)

$\rho 8$ si ripone il FF IBρ (GUN)

La lettura dei successivi caratteri in Memoria, il loro trasferimento ai FF FR e la registrazione su nastro continua con conta avanti dell'indirizzatore Y fino al carattere α di fine blocco.

Col carattere α di fine blocco si susseguono le seguenti operazioni:

Al mastro $\rho 0$ (M_4) il carattere α è disponibile nei FF Ud (Up) (U.C.)

$\rho 6$ (M_{10}) si ripone il FF βy con $\overline{\delta L}$.$\overline{\delta NAN}$.$\overline{IB}$. XNδ e cessa il pilotaggio della Memoria U.C. (U.C.)

$\rho 6$ (M_{10}) si ripone il FF Ce con $\overline{\beta y\delta}$ nell'U.C., si ripone il FF B_N, si riavvia la linea di ritardo e la si richiude su se stessa come si è visto per la LNa (U.C.)

	ρ7	il carattere α viene scritto su nastro	
	ρ8	si dispone il FF ∅Bρ	(GUN)

Nel p.d.c. successivo il GUN provvede a registrare su nastro il carattere NP.

Al mastro	ρ7	si registra su nastro il carattere NP	
	ρ8	si ripone il FF NPρ	(GUN)

Le operazioni nel GUN proseguono per la scrittura dei bits di sincronismo che seguono il carattere NP, dopo di che cessa il riciclo della linea ρ mentre l'arresto dell'U.N. in registrazione segue il segnale ∅B μ di fine blocco in verifica.

19.4. **Istruzione KN** (Xn IIII TP)

Decodificati ΔVEM, ΔN

Con questa istruzione si cancella il nastro magnetico per il tempo occorrente all'U.C. a percorrere le posizioni di Memoria che vanno dall'indirizzo IIII fino al primo carattere α. Il nastro si muove in avanti.

Preparazione dell'istruzione - Fase α

Unità centrale

Si opera come nell'istruzione RN.

G.U.N.

p.d.c. p_0 si dispone il FF Nf

p_1 si genera il segnale Qα che ripone il FF Vρ e i FF NR

p_2 si dispongono i FF ΔN, ZN ed R

p_7 si individua nel decodificatore U.N. l'U.N. interessata e si genera il segnale SN_k relativo ad essa

p_8 si genera il segnale SNR e si dispone quindi il FF NR_k che seleziona la testina di registrazione dell'U.N. interessata.

Quindi si interroga il FF No procedendo come nella RN.

Esecuzione dell'istruzione - Fase β

p.d.c. p_0 si azzerano gli indirizzatori W ed Y e vi si trasferisce il contenuto del registro 0. Si dispongono quindi i FF βy e βN (U.C.).

Si dispone al mastro M_5 il FF δKN con ΔVEM.βw.Nf.$\overline{No}$ (U.C.).

Si dispone al mastro M_8 il FF Ce con δKN.βw.P_0.ΔVEM. (U.C.).

Si ottiene nel GUN il segnale Qβ essendosi riposto il FF Nf.

Si dispone nell'U.C. il FF φ che pone termine alle operazioni che impegnano l'indirizzatore W.
Col segnale Qβ si succedono nel GUN le seguenti operazioni:

— si dispone il FF Vρ e si ottiene il segnale NA_k che dà inizio al moto del nastro avanti.

Diseccitandosi l'univibratore Uρ del GUN non si avvia la linea di ritardo ρ per la presenza del segnale ZN. Non inviando perciò alcun TrN all'U.C. la linea di ritardo di questa cicla col proprio p.d.c. D'altra parte il FF Ce è stato disposto in βw.P_0, e perciò il canale esterno entra immediatamente in funzione, senza attendere il tempo dato dall'univibratore Uρ, come avviene nella RN.

Ad ogni periodo di cifra l'U.C. invia al GUN dei caratteri, che però non vengono registrati per la mancanza dei mastri.
Leggendo in Memoria il carattere α, l'U.C. invia al GUN il decodificato ∅k; tale segnale determina l'arresto del moto avanti dell'U.N.

Nell'U.C. il segnale ∅k ripone al mastro M_{10} i FF δKN e βy; di conseguenza si ripongono i FF Ce e βN.

20. ISTRUZIONI DOPPIE IMPEGNANTI L'U.C.

20.1. **Istruzione PIN** (n_l n_s IIII TR)

Decodificati ΔVXB, ΔPI, ΔN

Questa istruzione precede la istruzione NAN.

Con essa si staticizza nell'indirizzatore Y l'indirizzo di Memoria IIII modificato dal contenuto del registro T.

Da questo indirizzo si parte nella successiva istruzione NAN per registrare in Memoria un blocco di caratteri provenienti da nastro magnetico.

Questa istruzione ha la durata di 10 p.d.c.

I caratteri specificati in P_7 e P_8 dell'istruzione individuano le U.N. che saranno usate nella successiva istruzione NAN.

Preparazione ed esecuzione dell'istruzione PIN

Fase α

Dura 9 p.d.c. e determina il trasferimento del carattere di funzione nei FF F e dell'indirizzo modificato nel registro J0.

Fase β

Dura un solo p.d.c. (p_0).

Nella fase β si svolgono le stesse operazioni elencate mastro per mastro nella istruzione PUM, procedendo al trasferimento nell'indirizzatore Y del contenuto del registro J0; quindi si dispongono al mastro M_7 il FF βy e al mastro M_9 il FF β_N.

Il trasferimento del contenuto del registro J0 all'indirizzatore Y è possibile solo se il FF βy è riposto.

Se il FF βy è disposto significa che il canale esterno è impegnato e in questo caso la istruzione PIN non viene eseguita.

In $\beta w.P_0$ si interroga il FF No del GUN. Tale FF si dispone se una delle unità n_s o n_l è occupata poichè in riavvolgimento; se il FF No è disposto l'istruzione PIN non viene eseguita.

Distinguiamo i seguenti casi:

a) in cui la istruzione PIN è posta sul 1° programma

b) in cui la istruzione PIN è posta sul 2° programma

c) in cui la istruzione PIN è posta sul programma« on line ».

a) **Istruzione PIN posta sul 1° Programma**

Se il canale esterno è occupato il FF βy è disposto; al mastro M_4 del p.d.c. p_5 della fase α dell'istruzione PIN si dispone il FF Ni. Se nel GUN si è disposto il FF No in $\beta w.P_0.M_4$ si dispone il FF Ni.

Le operazioni si susseguono come segue:

Fase β

p_0 M_0 si azzera l'indirizzatore W

M_4 si trasferisce all'indirizzatore W il contenuto del registro 0. (Il trasferimento all'indirizzatore Y è impedito dalla assenza del segnale $\overline{\beta y}$ per il primo caso; avviene ugualmente nel secondo caso)

M_7 si dispone il FF φ

M_7 si ripone il FF Kz

M_9 si azzera il contatore Cp. Il FF p_0 rimane disposto

M_9 si dispone il FF α

Fase α

p_0 M_0 si azzera l'indirizzatore W

M_4 si trasferisce all'indirizzatore W il contenuto del registro I

M_7 l'indirizzatore W non conta + 10

M_7 si ripone il FF Iw

M_9 il contatore Cp conta + 1. Il decodificatore di p-pulsi fornisce il segnale p_1

M_9 si ripone il FF p_0

p_1 M_0 l'indirizzatore W conta + 1

M_2 si dispone il FF Iw

M_2 si ripongono i FF F

M_3 si ripongono i FF Up (Ud)

M_4 si legge in Memoria estraendone di nuovo il carattere di funzione dell'istruzione PIN

M_7 si rigenera in Memoria il carattere letto al mastro M_4

M_9 il contatore Cp conta + 1.

Il decodificatore di p-pulsi fornisce il segnale p_2.

L'istruzione PIN ricicla finchè al p.d.c. p_0 della fase β non si ha il FF Ni riposto. Il FF Ni che, se disposto, determina il riciclo della istruzione PIN, si ripone in ogni fase α al p.d.c. p_3 e si dispone di nuovo al p.d.c. p_5 se il FF βy è disposto, o in $\beta w.P_0$ se è disposto il FF N_0.

b) **Istruzione PIN posta sul 2° Programma**

Se l'istruzione PIN è posta sul 2° programma, col FF α che dà inizio alla fase preparatoria, si dispone anche il FF α_2. Se il canale esterno è occupato e quindi il FF βy è disposto, al mastro M_4 del p.d.c. p_5 della fase α si attiva il FF Ne. Se nel GUN si è disposto il FF No in $\beta w.P_0.M_4$ si dispone il FF Ne.

Al p.d.c. p_0 della fase β si trasferisce all'indirizzatore W il contenuto del registro 0, mentre il trasferimento all'indirizzatore Y è impedito solo nella prima ipotesi dalla assenza del segnale $\overline{\beta y}$. Nello stesso p.d.c. p_0 si ripone al mastro M_9 il FF α_2 e si dispone il FF α.

Con la nuova fase α non si dispone più il FF α_2 perchè presente il segnale Ne. All'indirizzatore W verrà trasferito il contenuto del registro I anzichè quello del registro M e si passerà a preparare l'esecuzione di una istruzione del 1° programma.

Quando il FF βy si ripone o finisce il riavvolgimento che aveva determinato il disporsi del FF No, si dispone il FF Sb e, al termine di una n-esima istruzione di primo programma, si dispongono i FF α ed α_2. La presenza del segnale α_2 determina il ri-

torno al 2° programma e il trasferimento all'indirizzatore W del contenuto del registro M, cioè dell'indirizzo del carattere T della istruzione PIN.

Nella fase α non si avrà al mastro M_7 del p.d.c. p_0 la conta +10 poichè è ancora presente il segnale Ne che scompare in α, p_3, M_9 e quindi potrà essere ripetuta l'istruzione PIN.

c) **Istruzione PIN posta sul programma « on line »**

Quando l'istruzione PIN è posta sul programma « on line » (p.o.l.) si dispone col FF α anche il FF αp.

Se il canale esterno è occupato, cioè il FF βy è disposto, si attiva in α, p_5, M_4 il FF N_3. Se nel GUN si è disposto il FF No in $\beta w.P_0.M_4$ si dispone il FF N_3. Nel primo caso in β p_0 è impedito il trasferimento del contenuto del registro J0 all'indirizzatore Y, per l'assenza del segnale $\overline{\beta y}$.

Al mastro M_9, dello stesso p.d.c. si ripone il FF αp e si dispone il FF α (si suppone che il FF 2p sia riposto).

Con la nuova fase α si trasferisce all'indirizzatore W il contenuto del registro I anzichè quello del registro H e si opera quindi una istruzione del 1° programma.

Quando il FF βy si ripone o finisce il riavvolgimento che aveva determinato il disporsi del FF No, si dispone il FF Sb_3 e si riprende l'esecuzione del p.o.l. Nella prima fase α del p.o.l. che segue il disporsi del FF Sb_3, si opera di nuovo l'istruzione PIN trasferendo il contenuto del registro J0 all'indirizzatore Y.

20.2. **Istruzione NAN** ($n_l n_s$ IIII Tε)

Decodificati ΔVUR, ΔNN, ΔN

Con la istruzione NAN si opera un duplice trasferimento: da una zona di Memoria U.C. al nastro n_s e dal nastro n_l ad un'altra zona di Memoria U.C.

Nella fase α dell'istruzione si individua l'indirizzo di Memoria da cui si parte per il prelievo dei caratteri da registrare sul nastro n_s durante la fase esecutiva.

Questo indirizzo viene immagazzinato nel registro J0 durante la fase α e trasferito all'indirizzatore W al p.d.c. p_0 della fase β. L'indirizzatore Y contiene già l'indirizzo di Memoria, definito dalla precedente istruzione PIN, da cui si parte per registrare in Memoria i caratteri provenienti dal nastro n_l. Nella fase esecutiva sono occupati entrambi i canali della U.C. Nello svolgimento della istruzione NAN, la temporizzazione è affidata alla linea di ritardo del GUN, su cui si sincronizza la linea di ritardo della U.C.

Col sincronismo di lettura λ si opera il trasferimento dei caratteri letti sul nastro n_l alla Memoria U.C. e dalla Memoria U.C. alla Memoria σ-τ del GUN.

Il governo unità nastro provvede al trasferimento dei caratteri immagazzinati nella Memoria σ-τ al nastro n_s quando il contenuto della Memoria ha superato un valore prefissato.

I caratteri trasferiti sono quelli di un « blocco » e cioè contenuti fra 2 caratteri α di inizio blocco e di fine blocco.

Fase α

Durante la fase α si eseguono operazioni preparatorie nella U.C. e nel GUN. La temporizzazione è data dalla linea di ritardo della U.C. che cicla col proprio sincronismo.

Operazioni nella U.C.

p.d.c. p_0 si azzera l'indirizzatore W, vi si trasferisce il contenuto del registro I. L'indirizzatore W conta +10

p_1 l'indirizzatore W conta — 1.

Si legge in Memoria il carattere di funzione e lo si staticizza nei FF F

p_2 l'indirizzatore W conta — 1.

Si legge in Memoria il carattere che definisce l'indirizzo del registro T modificatore; esso viene scritto nell'indirizzatore V

p_3 l'indirizzatore W conta — 1.

L'indirizzatore V non conta. Si leggono i caratteri individuati in Memoria e nel registro T dai rispet-

tivi indirizzatori (W, V). Il risultato della somma dei due caratteri estratti, ed operati nella UAD, viene scritto nel registro J0

p_4 l'indirizzatore W conta — 1;
l'indirizzatore V conta + 1.

I caratteri individuati in Memoria e nel registro T vengono operati come al p.d.c. p_3 e il risultato viene scritto nel registro J0

p_5, p_6 } si opera come al p.d.c. p_4

p_7 l'indirizzatore W conta — 1;
l'indirizzatore V conta + 1.

I caratteri letti in Memoria e nel registro T vengono estratti e sommati nella U.A.D. come ai p.d.c. precedenti. Il risultato individua il nastro in registrazione n_s. Il risultato viene scritto nei FF Lu ed inviato al GUN

p_8 l'indirizzatore W conta — 1;
l'indirizzatore V non conta.

Manca il pilotaggio relativo al registro T. Si legge in Memoria il carattere che individua il nastro in lettura n_l. Esso viene scritto nei FF Ld ed inviato al GUN.

Operazioni nel G.U.N.

p.d.c. p_0 si dispone il FF Nf (nastro fermo). Si genera il segnale ZM, che scompare in βw, p_0 determinando l'azzeramento delle Memorie ρ-μ e σ-τ

p_1 si genera il mastro $Q\alpha$, che predispone taluni organi logici per l'esecuzione dell'istruzione

p_2 si dispongono i FF ΔN, Nan.

L'indicazione del FF di funzione Nan viene utilizzata in U.C.; essa permette che continui l'invio di caratteri da nastro a Memoria U.C. attraverso il canale esterno, anche quando è terminato il trasferi-

mento da Memoria U.C. a Memoria σ-τ e l'indirizzatore W è già passato all'esecuzione di istruzioni successive alla NAN, che impegnino i soli organi interni della U.C.

p_8 si dispone il FF NR_K in funzione del carattere inviato dall'U.C. al GUN in P_7; esso individua l'U.N. in registrazione.

Fase β

p.d.c. p_0 nella U. C. si azzera in M_0 l'indirizzatore W. In M_4 si trasferisce in W il contenuto del registro J0

nel G.U.N. si dispone il FF NL_K in base al carattere inviato dall'U.C. al GUN in P_8; il FF NL_K individua l'UN in lettura. Si ripone il FF Nf. Si ottiene il segnale NA_k che seleziona la testina di lettura e si dispone il FF Vλ che avvia il moto del nastro in lettura.

Nel tempo che intercorre tra l'inizio del moto del nastro in lettura e il primo « clock » la linea di ritardo della U.C. cicla col proprio sincronismo. In U.C. non ha luogo alcuna operazione per mancanza dei segnali di pilotaggio in Memoria ed assenza della conta degli indirizzatori W ed Y; si ha però la conta del contatore principale CP, ma ciò non ha alcuna importanza. Con la lettura del primo « clock » il GUN invia alla U.C. il primo segnale « Trig N » che ripone il FF Ie, aprendo la linea di ritardo della U.C. I successivi segnali « Trig N » generati dai « clock » letti su nastro sincronizzano la linea di ritardo della U.C. su quella λ del GUN; il mastro λm-1 coincide col mastro M_0.

Dopo i 3 bits di sincronismo Si si legge sul nastro magnetico il carattere NP (ᔕ).

Al mastro λo (M_5). il carattere NP è disponibile all'uscita dei FF $F_2\lambda$ del GUN e si genera il segnale PrN (pronto nastro in lettura) che viene inviato in U.C. Nello stesso p.d.c. si svolgono le seguenti operazioni:

Al mastro λ1 (M_6) si dispone il FF IB con $IWI.\overline{Ce}.PrN.$
$\beta y.\delta NAN.M_6$ (U.C.)

Al mastro λ3 (M_8) si dispone il FF Ce con $\overline{Ce}.PrN.\beta y.IB.$
$M_8.\delta NAN$ (U.C.)

Al mastro λ3 (M_8) si dispone il FF NPλ (GUN)

Al mastro λ3 (M_8) si dispone il FF IBλ (GUN)

Col segnale PrN è abilitata la conta dell'indirizzatore W e compare il segnale di consenso al pilotaggio in Memoria Ud (Up), col segnale Ce è abilitata la conta dell'indirizzatore Y e compare il segnale di consenso al pilotaggio in Memoria Up (Ud).

Nel p.d.c. successivo si legge in Memoria U.C. il carattere α di inizio blocco e lo si trasferisce alla Memoria σ-τ del GUN; si legge sul nastro magnetico il carattere α di inizio blocco e lo si trasferisce alla Memoria U.C.

Le operazioni si susseguono nel modo seguente, dove si è supposto che l'indirizzo iniziale dell'indirizzatore Y sia pari e quello dell'indirizzatore W dispari:

Mastro	M_0	(λm-1)	l'indirizzatore Y conta — 1 (IYP in IYD)
	M_0	(λm-1)	l'indirizzatore W conta — 1 (IWD in IWP)
	M_0	(λm-1)	si dispone il FF Zc (U.C.)
	M_2	(λm_1)	si dispone il FF Iy (su IYA) (U.C.)
	M_3	(λm_2)	si dispone il FF Zr (U.C.)
	M_4	(λm_3)	si legge in Memoria Up il carattere α e lo si staticizza nei FF Up
	M_5	(λ0)	il carattere α da nastro è disponibile nei FF $F_2\lambda$ del GUN
	M_7	(λ2)	si scrive in Memoria Ud il carattere α da nastro
	M_7	(λ2)	si rigenera in Memoria Up il carattere letto in M_4; è $XMG = \beta r.\Delta VUR$
	M_7	(λ2)	si ripone il FF Iw con $\Delta VUR.Ce.\overline{\alpha}.\overline{Ar}$ (U.C.)
	M_6	(λ1) o λ9	si trasferisce alla Memoria σ-τ del GUN il carattere α staticizzato nei FF Up

M_9	(λ4)	si genera nel GUN il segnale IBMi; esso ricorda che è arrivato l'inizio blocco da Memoria	
M_{10}	(λ5)	si ripone il FF IB	(U.C.)
	(λ8)	si ripone il FF IBλ	(GUN)

Nei successivi p.d.c. si operano i caratteri dei rispettivi blocchi in Memoria e sul nastro n_l con conta in avanti degli indirizzatori Y e W. All'arrivo del carattere α di fine blocco da nastro si ripone al mastro M_{10} (λ_5) il FF Ce con $\overline{IB}.\delta NAN.(XN\varepsilon+XN\eta)$. Col segnale $\overline{Ce}$ cessa il pilotaggio in Memoria per i caratteri provenienti da nastro e la conta dell'indirizzatore Y.

Al mastro λ_6 si dispone il FF $\varnothing$Bλ del GUN.

Nel p.d.c. successivo si legge sul nastro il carattere NP (∽) che non viene scritto in Memoria U.C. per assenza del segnale di pilotaggio.

Al mastro λ_6 del p.d.c. relativo alla lettura del carattere NP si dispone il FF $\varnothing$Nnλ del GUN che provvede da questo istante in avanti a fare ciclare su se stessa la linea di ritardo λ e a fornire il segnale « Trig N » necessario per la sincronizzazione della linea di ritardo della U.C. su quella λ del GUN; il segnale $\varnothing$Nnλ assicura inoltre la permanenza del segnale PrN.

Al mastro λ_6 si ripone nel GUN il FF NPλ.

L'UN n_l (in lettura) si arresta, mentre quella in registrazione continua il suo moto.

All'arrivo del carattere α di fine blocco da Memoria al mastro λ_4 scompare il segnale IBMi e compare IBMφ; al mastro λ_7 si ripone il FF $\varnothing$Nnλ con IBMφ; di conseguenza scompare il segnale PrN.

Col segnale PrN cessa la conta dell'indirizzatore W e il segnale di consenso al pilotaggio in Memoria. Con la lettura del carattere α in Memoria si dispone al mastro M_7 il FF φ con $\Delta VUR.\beta w.\overline{IB}.XN\delta$ e si ripone al mastro M_{10} il FF βy con $\Delta VUR.\varphi.\overline{Ce}$.

Disponendosi il FF φ hanno termine le operazioni che impegnano nella istruzione NAN il canale interno. L'U.C. può iniziare la preparazione dell'istruzione successiva alla NAN.

Se il carattere α da Memoria arriva prima del carattere α da nastro, ha termine il trasferimento dei caratteri da Memoria U.C. a Memoria σ-τ, mentre continua quello da nastro a Memoria U.C. Si dispone pertanto il FF φ e l'elaboratore può iniziare la preparazione dell'istruzione successiva alla NAN; questa potrà essere eseguita solo se non impegna il canale esterno ancora occupato.

Se invece l'istruzione letta impegna il canale esterno essa viene riciclata finchè non si ripone il FF βy. Con la lettura su nastro del carattere α di fine blocco si genera il segnale XNε che ripone al mastro M_{10} il FF βy con $\delta NAN.\overline{IB}.X\beta y.(XN\varepsilon+XN\eta)$; è $X\beta y=\Delta VUR.\overline{\alpha_r}.\overline{PoR}.\overline{XN\lambda}.\overline{XN\Theta}$; si ha perciò $X\beta y$, e quindi non si permette il riporsi di βy all'arrivo del fine blocco da nastro in una Nan, se non sono ancora finite le operazioni esecutive relative al canale interno $\Delta VUR.\overline{\alpha_r}.\overline{P_0}R)$ e non è presente il fine blocco da Memoria $(\overline{XN\lambda}.\overline{XN\Theta})$.
Con $\overline{\beta y\delta}$ si ripone il FF Ce.

Se il carattere α da Memoria arriva contemporaneamente a quello da nastro il FF βy si ripone come nell'ultimo caso esaminato; proprio per questo caso particolare nell'and che genera il segnale $\overline{X\beta y}$ si ha $\overline{XN\lambda}.\overline{XN\Theta}$.

Il FF Ce si ripone sia con $\beta y\delta$ che con $\Delta NAN.\overline{IB}.(XN\varepsilon+XN\eta)$.

Il FF φ si dispone come nel primo caso esaminato.

Se gli indirizzi iniziali dei contatori W e Y sono entrambi pari o dispari, leggendo su nastro il carattere NP che precede il blocco al mastro M_8 (λ_3) si dispone in U.C. il FF Ar con $IB.\beta y.\overline{Ce}.PrN$. Il segnale Ar impedisce la conta dell'indirizzatore W ed il relativo pilotaggio di Memoria. Inoltre, per la presenza di Ar, si ha il segnale $\delta PRm = \Delta VUR.\beta w.\overline{P_0}.Ar$; tale segnale impedisce che si abbia nel GUN la conta dell'indirizzatore di scrittura della Memoria σ-τ e la relativa scrittura.

Istruzione NDN (n_l n_r IIII Tβ)

Decodificati $\Delta V\Sigma R$, ΔN

Con la istruzione NDN si opera il trasferimento di un blocco di caratteri da Memoria U.C. al nastro magnetico n_s, e contemporaneamente si registra in Memoria U.C. un blocco di caratteri letto sul nastro magnetico n_l.

Le posizioni di Memoria in cui si scrivono i caratteri provenienti dal nastro n_l sono quelle dalle quali sono stati prelevati i caratteri diretti al nastro n_s. I due blocchi in Memoria e su nastro magnetico hanno la stessa lunghezza.

I caratteri vengono letti e registrati a gruppi compresi fra speciali caratteri Θ intercalati nel blocco. Sul nastro si hanno due caratteri Θ tra un gruppo di caratteri e l'altro mentre in Memoria ve ne è uno solo. Essi si presentano nel modo seguente:

blocco su nastro: $\alpha\ \Theta\Theta \cdots \Theta\ \Theta \cdots \Theta\ \Theta \quad \Theta\ \Theta\ \alpha$
blocco in Memoria: $\alpha\ \Theta \cdots \Theta \cdots \Theta \quad \Theta\ \alpha$

I caratteri compresi fra due caratteri Θ individuano di solito il contenuto di una scheda meccanografica.

In corrispondenza al secondo carattere Θ di ogni gruppo letto su nastro è possibile cambiare l'indirizzo di Memoria da cui si iniziano le operazioni relative a un gruppo di caratteri, in base alle indicazioni di una direttrice il cui indirizzo iniziale è quello indicato dall'istruzione: IIII modificato dal contenuto del registro T.

L'istruzione NDN impegna entrambi i canali della U.C., e quindi non possono essere eseguite altre istruzioni contemporaneamente.

Fase α

Durante la fase α si dispongono gli organi interni della U.C. e del GUN per l'esecuzione dell'istruzione.

Elenchiamo le principali operazioni che avvengono nella U.C. e nel GUN durante la fase α di preparazione:

Nella U.C.

p.d.c. $p_0\ M_0$ si azzera l'indirizzatore W

$p_0\ M_4$ si trasferisce all'indirizzatore W il contenuto del registro I

$p_0\ M_7$ l'indirizzatore W conta + 10

$p_1\ M_0$ l'indirizzatore W conta — 1

$p_1\ M_4$ si trasferisce da Memoria ai FF F il carattere di funzione

p_2 M_0 l'indirizzatore W conta — 1
Si azzera l'indirizzatore V

p_2 M_4 si legge in Memoria e si scrive nell'indirizzatore V l'indirizzo del registro T modificatore

p_3, p_4, p_5, p_6 — Ad ogni p.d.c. si legge in Memoria e si trasferisce nel registro J0 il carattere letto in Memoria e modificato dal contenuto del registro T

p_7 — Si trasferiscono da Memoria ai FF Lu, Ld i caratteri che individuano i due nastri nr, nl.

Questi vengono contemporaneamente inviati al decodificatore UN del GUN che seleziona i nastri in lettura e registrazione.

p_8 — Il carattere letto in Memoria al p.d.c. p_7 viene modificato dal contenuto del registro T.

Col p.d.c. p_8 termina la fase α. Al mastro M_8 si ripone il FF Kz. Al mastro M_9 si ripone il FF α, si dispongono i FF p_0 e βw, si azzera il contatore Cp.

Nel GUN

p.d.c. p_0 si dispone il FF Nf e si genera il segnale ZM, che scomparendo in βw, p_0, M_9, determina l'azzeramento delle Memorie ρ-μ e σ-τ

p_1 si genera il segnale Qα, che predispone taluni organi logici per l'esecuzione della istruzione

p_2 si dispone il FF ΔN.
Si dispongono i FF Ndn e Ndnm.
Il segnale Ndnm viene inviato dal GUN alla U.C. e nell'U.C. assume il nome δNDN

p_8 si dispone il FF NR_k in funzione del contenuto dei FF Lu, esso individua l'U.N. in registrazione.

Si interroga il FF No. Se il FF No si dispone l'istruzione non viene eseguita e ricicla in U.C. finchè non scompare il segnale No.

Fase esecutiva β

Operazioni in U.C. e nel GUN

p.d.c. p_0 — si azzerano gli indirizzatori W ed Y (U.C.)
si dispone in funzione del contenuto dei FF Ld, il FF NL_k (GUN) che individua l'unità in lettura (selezione delle testine di lettura)

si interroga il FF No (GUN).

Se il FF No si dispone, la istruzione non può essere eseguita e ricicla nella U.C. Se il FF No non si dispone, la istruzione può proseguire nel suo svolgimento

p_0 M_4 — si trasferisce il contenuto del registro J0 (U.C.) agli indirizzatori W ed Y (U.C.) e lo si riscrive nel registro J0 al mastro M_7

p_0 M_5 — si dispone il FF Kz (U.C.)

p_0 M_7 — si dispone il FF βy (U.C.)

p_0 M_8 — si ripone il FF Kz con $\Delta N.\Delta\Sigma.\overline{\mu}.\beta w.P_0$ (U.C.)

p_0 M_9 — si dispone il FF μ con $\Delta N.\Delta\Sigma.P_0.S\beta y$ (U.C.)

p_0 M_9 — si ripone il FF Nf (GUN)
si genera il segnale Qβ (GUN).

Col segnale Qβ si susseguono nel GUN le operazioni seguenti:
Si dispone il FF Vλ e si ottiene il segnale NA_k che dà inizio al moto del nastro in lettura.

Si dispongono i FF Slρ ed Slμ che condizionano la selezione delle testine di registrazione e di lettura in verifica.

L'unità nastro in lettura inizia il suo moto in avanti, mentre quella in registrazione parte solo quando nel trasferimento dei caratteri letti in Memoria U.C. alla Memoria σ-τ del GUN, si è disposto il FF C_{10} del contatore differenziale Cδ (contenuto di Cδ >1663).

Nel periodo che intercorre tra la partenza del nastro in lettura e la lettura del primo bit di sincronizzazione Siλ, l'U.C. prosegue nella preparazione della istruzione NDN.

Il contatore Cp della U.C. è ancora azzerato e il FF p_0 ancora disposto, per cui, al successivo riciclo della linea di ritardo, si opera un secondo periodo di cifra p_0 con FF μ disposto.

1ª Fase μ, sovrapposta alla fase β

p_0 M_0	si azzera l'indirizzatore Y
p_0 M_3	si dispone il FF μ_r
p_0 M_4	si trasferisce il contenuto del registro J0 all'indirizzatore Y; sono AJO = $\mu.\overline{P_1}.\overline{P_2}$, CJY = $\mu.P_0$
p_0 M_5	si dispone il FF Kz (U.C.)
p_0 M_9	si ripone il FF p_0 (U.C.)
p_0 M_9	il contatore Cp conta + 1
p_1	L'indirizzatore Y non conta mancando il segnale Ce; è pure assente il relativo pilotaggio.
p_2	La conta dell'indirizzatore W è impedita da $\mu.P_1$ e $\mu.P_2$; anche il relativo pilotaggio è impedito per la presenza di XMY
p_3 M_0	l'indirizzatore W conta — 1; è presente il relativo pilotaggio
p_3 M_4	si legge in Memoria e si trasferisce il carattere letto ai FF Ud (Up)
p_3 M_7	si scrive nel registro J0 unità il carattere letto in Memoria all'M_4.
	Esso viene rigenerato in Memoria U.C. essendo presente XMG = μ_r
p_4	si opera come al p.d.c. p_3, nel registro 0 decine
p_5	si opera come al p.d.c. p_3, nel registro 0 centinaia
p_6 M_0	l'indirizzatore W conta — 1
p_6 M_4	si legge in Memoria all'indirizzo specificato dall'indirizzatore W e si staticizza il carattere letto nei FF Ud (Up)
p_6 M_7	si trasferisce il carattere letto all'M_4 nel registro 0 migliaia, rigenerandolo in Memoria U.C.

p_6 M_8 si ripone il FF Kz con $\mu.P_6.\overline{\lambda}$

p_6 M_9 si dispone il FF λ con $\mu.P_6$

p_6 M_9 si dispone il FF p_0

p_6 M_9 si azzera il contatore Cp.

È stato così trasferito nel registro J0 il primo indirizzo della direttrice.

2ª Fase μ, sovrapposta alla fase β

p_0 M_0 si azzera l'indirizzatore Y

p_0 M_4 si trasferisce all'indirizzatore Y il contenuto del registro J0, cioè il primo indirizzo della direttrice

p_0 M_5 si dispone il FF Kz

p_0 M_9 si ripone il FF p_0

p_0 M_9 il contatore Cp conta + 1

p_1, p_2, p_3, p_4, p_5 } si opera come ai p.d.c. p_1, p_2, p_3, p_4, p_5 della precedente fase $\mu + \beta$

p_6 M_0 l'indirizzatore W conta — 1

p_6 M_4 si legge in Memoria all'indirizzo specificato dall'indirizzatore W e si staticizza il carattere letto nei FF Up (Ud)

p_6 M_7 si trasferisce il carattere letto al mastro M_4 nel registro 0 e lo si rigenera in Memoria

p_6 M_9 si ripone il FF μ con $\mu.\lambda.P_6$.

È stato così trasferito nel registro J0 il secondo indirizzo della direttrice.

Da questo istante fino all'arrivo del primo carattere letto su nastro l'U.C. non esegue alcuna operazione, essendo impedita la conta degli indirizzatori W ed Y e mancando il pilotaggio in Memoria. La conta dell'indirizzatore W è impedita da $\delta NDN.\Delta N.\Delta\Sigma.\beta y.\overline{\mu}$; il relativo pilotaggio dalla presenza di $XMY = \overline{\mu}.\Delta N.\Delta\Sigma$.

Quando il primo « clock » inciso sul nastro n_1 viene letto dalle testine di lettura, si genera nel GUN il segnale « Trig N » che ripone il FF Ie, interrompendo il riciclo della linea di ritardo della U.C. Il segnale « Trig N » viene generato ad ogni mastro λm-1 della linea di ritardo λ del GUN e sincronizza la linea di ritardo U.C. su quella λ. ($M_0 \equiv \lambda m\text{-}1$).

Il primo carattere letto su nastro è il carattere NP (∽); con esso hanno luogo le seguenti operazioni:

Al mastro	λ_0 (M_5)	si genera il segnale Prn	(GUN)
	λ_3 (M_8)	si dispone il FF NP λ	(GUN)
	λ_3 (M_8)	si dispone il FF IB λ	(GUN)
	λ_3 (M_8)	si dispone il FF Ce con $PrN.\overline{Ce}.\overline{\delta NAN}.\beta y$	(U.C.)
	λ_3 (M_8)	si dispone il FF IB con $PrN.\overline{Ce}.\overline{\delta NAN}.\overline{\delta PRC}.\overline{\mu}.\beta y$	(U.C.)

Con i segnali PrN e Ce si ottengono i segnali di consenso al pilotaggio connesso all'indirizzatore Y delle Memorie Mp, Md in U.C.

Alla lettura su nastro del carattere NP segue quella del carattere α su nastro e del carattere α in Memoria U.C.

Al mastro	M_0 (λm-1)	l'indirizzatore Y conta — 1	(U.C.)
	M_2 (λm_1)	si dispone il FF Iy su IYA	(U.C.)
	M_4 (λm_3)	si legge in Memoria U.C. e si trasferisce il carattere letto ai FF di uscita Ud (Up)	(U.C.)
	M_5 (λ0)	il carattere α letto su nastro è staticizzato nei FF $F_2\lambda$	(GUN)
	M_7 (λ2)	si scrive in Memoria U.C. il carattere α da nastro e in Memoria σ-τ del GUN il carattere α da U.C.	
	M_{10}(λ5)	si ripone il FF IB	(U.C.)
	λ8	si ripone il FF IB λ	(GUN)

Al carattere α può seguire su nastro e in Memoria U.C. il carattere Θ.

M_0 (λm-1)	l'indirizzatore Y conta + 1	(U.C.)

M_4 (λm3) si legge in Memoria U.C. e si trasferisce il carattere letto ai FF di uscita Up (Ud) (U.C.)

M_5 (λ0) il carattere Θ da nastro viene scritto nei FF F_2 λ (GUN); tale carattere genera in U.C. il segnale IΘ

M_7 (λ2) si scrive in Memoria U.C. il carattere Θ da nastro e in Memoria σ-τ del GUN il carattere Θ da U.C.

M_8 (λ3) si ripone il FF Kz con IΘ (U.C.)

M_9 (λ4) si dispone il FF μ con IΘ (U.C.)
si dispone il FF Po (U.C.)
si azzera il contatore Cp (U.C.)

M_{10}(λ5) si ripone il FF Ce con IΘ (U.C.)

Nel p.d.c. p_0 successivo si legge su nastro il secondo carattere Θ che non viene scritto in Memoria U.C. per mancanza del pilotaggio data l'assenza di Ce; in questo p.d.c. la Memoria σ-τ non riceve quindi alcun carattere da quella di U.C.

3ª Fase μ, sovrapposta alla fase β

p.d.c. p_0 M_0 si azzera l'indirizzatore Y (U.C.)

p_0 M_4 si trasferisce all'indirizzatore Y il contenuto del registro 0 e cioè il secondo indirizzo della direttrice, ottenuto nella 2ª fase μ (U.C.)

p_0 M_4 si ripone il FF Iy su IYI con $\beta y.\overline{Ce}$ (U.C.)

p_0 M_5 si dispone il FF Kz (U.C.)

p_0 M_8 si dispone il FF Ce con $\overline{\delta NAN}.PrN.\beta y.\overline{Ce}$ (U.C.)

p_0 M_9 si ripone il FF Po (U.C.)

p_0 M_9 il contatore Cp conta + 1 (U.C.)

p.d.c. p_1 M_0 l'indirizzatore Y conta — 1;
l'indirizzatore W non conta (U.C.)

p_1 M_2 si dispone il FF Iy (U.C.)

p_1 M_4 si legge in Memoria U.C. all'indirizzo indicato da Y e si trasferisce il carattere letto ai FF Up (Ud)

p_1 M_5 ($\lambda 0$) si trasferisce ai FF F_2 λ del GUN il carattere letto su nastro

p_1 M_7 ($\lambda 2$) si scrive in Memoria U.C. il carattere proveniente da nastro e in Memoria σ-τ quello estratto da Memoria U.C.

p_2 si opera come al p.d.c. precedente con conta avanti dell'indirizzatore Y

p_3 M_0 l'indirizzatore Y conta + 1; l'indirizzatore W conta — 1. (U.C.)

p_3 M_4 si legge in Memoria U.C. all'indirizzo indicato da Y e si trasferisce il carattere letto ai FF Up (Ud)

p_3 M_4 si legge in Memoria U.C. all'indirizzo individuato da W e si trasferisce il carattere letto ai FF Ud (Up)

p_3 M_5 si trasferisce ai FF F_2 λ del GUN il carattere letto su nastro

p_3 M_7 ($\lambda 2$) si scrive in Memoria U.C. all'indirizzo indicato da Y il carattere proveniente da nastro e si trasferisce alla Memoria σ-τ del GUN il contenuto dei FF Up (Ud).

Si rigenera in Memoria UC all'indirizzo indicato da W il carattere letto in M_4 e si trasferisce il contenuto dei FF Ud (Up) al registro J0 (U.C.)

p_4, p_5 } si opera come al p.d.c. p_3

p_6 si opera come al p.d.c. p_3 fino al mastro M_7.

Al mastro M_9 si ripone il FF μ.

Terminata la fase μ contemporanea alla β, proseguono le operazioni relative al canale esterno fino alla lettura del carattere Θ che

delimita un gruppo di caratteri all'interno del blocco. Il carattere Θ avvia una fase μ in cui si opera, nel modo precedentemente descritto, il trasferimento del contenuto del registro 0 all'indirizzatore Y e di un nuovo indirizzo di Memoria al registro 0.

Le operazioni dell'istruzione NDN che impegnano l'U.C. hanno termine con la lettura del carattere α su nastro contemporanea a quella del carattere α in U.C. Con la lettura del carattere α si genera al mastro $\lambda 0$ (M_5) il segnale $XN\rho$ col carattere letto su nastro, e al mastro M_4 il segnale $XN\delta$ col carattere letto in Memoria U.C. Si ottiene pertanto il segnale $XF\mu = \Delta N \,.\, \Delta\Sigma \,.\, XN\rho + \Delta N . \Delta\Sigma . XN\delta$.

Al mastro M_7 ($\lambda 2$) si dispone il FF φ con $XF\mu.\beta w.\overline{IB}$ (U.C.)

Al mastro M_{10} ($\lambda 5$) si ripone il FF βy con $\overline{\delta NAN}.\overline{IB}.XN\rho$; oppure con

$XN\delta.\overline{IB}.\overline{\delta L}.\overline{\delta NAN}$ (U.C.)

Al mastro M_{10} ($\lambda 5$) si ripongono i FF Ce e βN con $\overline{\beta y \delta}$ (U.C.)

Al mastro $\lambda 6$ si dispone il FF $\varnothing B\lambda$ (GUN)

Col segnale $\overline{\beta y \delta}$ si riavvia la linea di ritardo di U.C.; al successivo M_2 si dispone il FF Ie e la linea di ritardo si richiude su se stessa.

Ulteriori trigger da nastro non possono influire per l'assenza del segnale βN.

Se quando si dispone il FF φ è in corso una fase μ, il FF μ si ripone e la fase stessa viene interrotta; infatti essa non ha alcuna utilità.

Mentre l'U.C. inizia la preparazione di una nuova istruzione, il GUN continua il trasferimento al nastro n_s dei caratteri contenuti nella Memoria σ-τ finchè questa non è azzerata.

L'istruzione NDN può essere « parzializzata », cioè è possibile porre « inesistente » una delle due unità nl o ns. Si hanno in tali casi le seguenti differenze nel funzionamento.

Unità inesistente (carattere #) in lettura

La fase α è identica a quella già descritta.

In βw, P_0 si ripone il FF Ndn e si dispone il FF R (G.U.N.); cioè per quanto riguarda il GUN l'istruzione NDN diventa una registrazione.
Col segnale $Q\beta$ si avvia l'unità in registrazione procedendo esattamente come in una semplice registrazione.

Nel periodo che intercorre tra la partenza del nastro in registrazione e l'inizio del primo segnale « Trig N » all'U.C. si effettuano la 1ª e la 2ª fase μ come già descritto precedentemente.

Quando il primo « clock » viene registrato sul nastro n_s si genera nel GUN il segnale « Trig N ». Tale segnale viene generato ad ogni mastro ρm della linea di ritardo ρ del GUN e sincronizza la linea di ritardo U.C. ($M_0 \equiv \rho m$).

Si procede quindi esattamente come già descritto; l'unica differenza è che il segnale $I\Theta$ non può essere ottenuto con il carattere Θ letto su nastro; la presenza del carattere # in P_8 dell'istruzione, e quindi nel registro lunghezze decine genera il segnale $\Delta\eta L_{10}$, con cui si ottiene il segnale $I\Theta$ ogni qualvolta si legge un Θ nella Memoria di U.C.

Il FF Ndm (GUN), oltre che essere inviato all'U.C., serve anche nel GUN stesso per ricordare che la registrazione è in realtà una NDN parzializzata, e che quindi occorre raddoppiare i caratteri Θ.

Il motivo del raddoppio di tali caratteri è ovviamente dovuto al fatto che fra una scheda e la successiva occorre interporre un p.d.c. per permettere il cambio di indirizzo nel contatore Y. In tale periodo di cifra il canale esterno non è utilizzabile.

Unità inesistente (carattere #) in registrazione

In α, P_8 si ripone il FF Ndn e si dispone il FF L (G.U.N.); l'istruzione, per quanto riguarda il GUN, diventa una lettura.

Il segnale « Trig N » viene generato ad ogni mastro λm della linea di ritardo λ del GUN ($M_0 \equiv \lambda m$). Vi è quindi uno sfalsamento di un mastro rispetto alla NDN non parzializzata; per il resto tutto avviene come nella NDN completa.

È qui opportuno calcolare quale è il minimo numero di caratteri che deve contenere una scheda affinchè sia possibile effettuare una fase μ nel corso della lettura della scheda stessa; teniamo presente che in $\mu.P_0.M_8$ possono essere entrambi gli indirizzatori W ed Y pari o dispari; quindi si può avere un doppio P_0, cioè P_0 seguito da P_0c; si avrà comunque sempre un doppio P_1 e un doppio P_2, dato che l'indirizzatore W non conta in P_1 e P_2.

In definitiva la fase μ occupa 9 o 10 periodi di cifra. Dato che su nastro al primo Θ che dispone il FF μ segue un secondo Θ, i caratteri di una scheda compresi fra i Θ devono essere 9 per il caso più sfavorevole.

21. ISTRUZIONI CHE NON IMPEGNANO L'U.C.

21.1. Istruzione PRN (n_l n_r IIII T O)

Decodificati ΔVUM, ΔN

Si deve notare che il codice di funzione è la lettera O; in realtà con la presente istruzione si impegna l'U.C. anche durante l'esecuzione; è stata posta nel presente capitolo dato che deve essere abbinata all'istruzione TN.

Con la istruzione PRN si opera il trasferimento dalla Memoria dell'U.C., a partire dall'indirizzo specificato nell'istruzione e per indirizzi crescenti, alla Memoria χ del GUN, i caratteri comprendenti la parola chiave che sarà usata nella ricerca su nastro.

Fase α

Elenchiamo le principali operazioni che avvengono durante la fase α di preparazione:

Nella U.C.

p.d.c. p_0 si azzera l'indirizzatore W, si trasferisce in esso il contenuto del registro I; si conta + 10

p_1 l'indirizzatore W conta —1; si trasferisce da Memoria ai FF F il carattere di funzione

p_2 l'indirizzatore W conta — 1; si azzera l'indirizzatore V; si legge in Memoria e si scrive nell'indirizzatore V l'indirizzo del registro modificatore.

p_3, p_4, p_5, p_6 } Ad ogni p.d.c. si legge in Memoria e si trasferisce nel registro 0 il carattere letto in Memoria e modificato dal contenuto del registro T

p_7, p_8 } si trasferiscono da Memoria ai FF Lu, Ld i caratteri che individuano i due nastri n_r, n_l tra i quali dovrà essere effettuata la successiva istruzione con ricerca. Tali caratteri sono anche inviati al GUN.

Col p.d.c. p_8 termina la fase α. Al mastro M_8 si ripone il FF Kz. Al mastro M_9 si ripone il FF α, si dispongono i FF p_0 e βw, si azzera il contatore Cp.

Nel GUN

p.d.c. p_0 si dispone il FF Nf

p_1 si genera il segnale $Q\alpha$

p_2 si dispone il FF ΔN ed il FF Prc; il segnale Prc è inviato all'U.C., dove è denominato δPRC

p_8 col carattere inviato all'U.C. in p_7, e cioè il carattere che individua il nastro n_r, si interroga il FF No. Se il FF No si dispone l'istruzione non viene eseguita; ciò perchè non potendosi eseguire la successiva TN non può essere eseguita neppure la PRN.

Fase esecutiva β

Operazioni in U.C. e nel GUN

p.d.c. p_0 si azzera l'indirizzatore Y (U.C.)
col carattere inviato dall'U.C. in p_8 si interroga il FF No (GUN);

se il FF No si dispone l'istruzione non viene eseguita;

si trasferisce il contenuto del registro 0 all'indirizzatore Y (U.C.)

si dispone il FF βy (U.C.)

si dispone il FF φ (U.C.)

si ripone il FF Nf e si genera il segnale $Q\beta$ (GUN)

si dispone il FF βN (U.C.)

Col disporsi del FF φ hanno termine in U.C. le operazioni che impegnano l'indirizzatore W; l'U.C. può quindi eseguire istruzioni che impegnano il canale interno.

Il segnale $Q\beta$ avvia la linea di ritardo λ del GUN; tale linea di ritardo viene riavviata dal λ 8; si ottiene quindi un riciclo di 12 μ sec. della linea di ritardo λ del GUN.

Col mastro λm viene inviato all'U.C. il segnale « Trig N » che sincronizza la linea di ritardo della U.C. su quella λ del GUN ($\lambda m = M_0$).

Nel GUN si hanno tre cicli di mastri di preparazione; viene usata la stessa logica adoperata per la registrazione su nastro dei primi tre « clocks ».

Al mastro λ1 (M_5) del quarto ciclo il GUN invia all'U.C. il segnale PrN.

Al mastro M_8 (λ4) si dispone il FF Ce (U.C.).

Durante il successivo p.d.c.

M_0	(λm)	l'indirizzatore Y conta — 1 (U.C.) è presente il consenso al pilotaggio della Memoria di U.C.
M_2	(λm_2)	si dispone il FF Iy su IYA
M_4	(λ0)	il primo carattere della parola chiave per la ricerca è disponibile nei FF Up (Ud) e viene inviato al GUN (U.C.)
M_7	(λ3)	si rigenera in Memoria U.C. il carattere letto in M_4
M_8	(λ4)	si scrive nella Memoria λ del GUN il carattere letto in M_4.

I successivi periodi di cifra sono identici con conta avanti dell'indirizzatore Y.

Nell'ultimo p.d.c. si legge nella Memoria U.C. il carattere ⊗; il GUN invia all'U.C. al mastro λ0 (M_4) il segnale ∅ PRC; con tale segnale al mastro M_{10} si ripone il FF βy, quindi il FF Ce.

21.2. **Istruzione TN** (n_l n_r BBBB T $)

Decodificati ΔVEB, ΔN

Con tale istruzione si ordina al GUN di trascrivere dal nastro in lettura n_l al nastro in registrazione n_r il numero di blocchi, eventualmente modificato, specificato nell'istruzione.

Se l'istruzione TN è preceduta dall'istruzione PRN è inutile specificare il numero di blocchi; la trascrizione infatti procede finchè si trova un blocco con chiave uguale o superiore a quella trasferita nella Memoria del GUN con l'istruzione PRN.

L'istruzione TN occupa l'U.C. solo durante la fase α di preparazione e per un periodo di cifra (P_0) della fase βw.

Elenchiamo le principali operazioni:

Nella U.C.

p.d.c. p_0 si azzera l'indirizzatore W; si trasferisce in esso il contenuto del registro I; si conta + 10

p_1 l'indirizzatore W conta — 1; si trasferisce da Memoria ai FF F il carattere di funzione

p_2 l'indirizzatore W conta — 1; si azzera l'indirizzatore V; si legge in Memoria e si scrive nell'indirizzatore V l'indirizzo del registro modificatore

p_3, p_4, p_5, p_6 } Ad ogni p.d.c. si legge in Memoria e si invia al GUN il carattere letto in Memoria e modificato dal contenuto del registro T; tali caratteri individuano il numero di blocchi che devono essere trascritti

p_7, p_8 } si trasferiscono da Memoria al GUN i caratteri che individuano i due nastri n_r, n_l tra i quali dovrà essere effettuata la trascrizione.

Col p.d.c. p_8 termina la fase α.

p.d.c. βw, p_0 si dispone al mastro M_7 il FF φ; si ripone il FF βw; hanno così termine le operazioni che impegnano l'U.C., questa può perciò eseguire istruzioni che impegnino sia il canale interno che il canale esterno.

Nel G.U.N.

Si distinguono i seguenti casi:

a) **Istruzione TN preceduta da PRN**

Dato che la TN deve seguire immediatamente la PRN essa viene letta durante l'esecuzione della PRN, ma, non potendo essere eseguita, viene riciclata finchè la PRN non è finita.

p.d.c. α, p_0 si dispone il FF Nf

p_1 si genera il segnale $Q\alpha$

p_2 si dispone il FF ΔN ed il FF T

p_3 si dispone il FF Rc

p_6 si ripone il FF Prc

p_8 si dispone il FF NR_k corrispondente all'unità a nastro n_r; se tale unità è inesistente, cioè al posto di n_r è stato scritto il carattere # volendosi fare una ricerca senza trascrizione, si ripone il FF T

p.d.c. β, p_0 si dispone il FF NL_k corrispondente all'unità a nastro n_l; tale unità non può essere inesistente; si ripone il FF Nf e si genera il segnale Qβ che dà inizio all'esecuzione dell'istruzione.

b) **Istruzione TN non preceduta da PRN**

p.d.c. α, p_0 si dispone il FF Nf

p_1 si genera il segnale Qα

p_2 si dispone il FF T

p_3, p_4, p_5, p_6 i caratteri inviati dall'U.C. vengono staticizzati nelle rispettive decadi del contatore di blocchi del GUN

p_8 si dispone il FF NR_k corrispondente all'unità a nastro n_r, si interroga il FF No; se l'unità n_r risulta occupata l'istruzione non può essere eseguita

p.d.c. β, p_0 si dispone il FF NL_k corrispondente all'unità a nastro n_l; si interroga il FF No; se l'unità n_l risulta occupata l'istruzione non può essere eseguita; in tal caso si dispone il FF Ni (se l'istruzione TN è stata letta sul 1° programma).

Se l'istruzione può essere eseguita si ripone il FF Nf e si genera il segnale Qβ che dà inizio all'esecuzione.

Se in questo secondo caso l'unità in registrazione n_r è inesistente, per il GUN la TN diventa una DUB; infatti in α, p_8 si ripone il FF T e si dispone il FF D.

Se l'unità in lettura è inesistente, per il GUN la TN diventa una KN, cioè una cancellazione, fino al fine nastro; infatti in βw, p_0 si ripone il FF T e si dispongono i FF ZN ed R.

21.3. **Istruzione DUB** (n BBBB T N)

Decodificati ΔVUB, ΔN

Con tale istruzione si ordina al GUN di far procedere il nastro n avanti o indietro di un numero di blocchi pari a quelli specificati nell'istruzione.

Il carattere specificato in p_7, eventualmente modificato, per quanto riguarda i bit a, b, c specifica le decine di migliaia di blocchi; quindi al massimo può avere il valore 4; se è presente il bit d il nastro procede all'indietro, se è assente procede in avanti.

L'istruzione DUB occupa l'U.C. solo durante la fase α di preparazione e per un periodo di cifra (p_0) della fase βw.

Elenchiamo le principali operazioni:

Nella U.C.

p.d.c. α, p_0, p_1, p_2, p_3, p_4, p_5, p_6 tutto avviene come nella TN

p_7 si trasferisce da Memoria al GUN il carattere che individua le decine di migliaia di blocchi e il senso del moto dell'unità a nastro

p_8 si trasferisce da Memoria al GUN il carattere che individua il nastro n.
Ha termine la fase α

p.d.c. β, p_0, si dispone al mastro M_7 il FF φ; si ripone il FF βw; hanno così termine le operazioni che impegnano l'U.C.

Nel G.U.N.

p.d.c. α, p_0 si dispone il FF Nf

p_1 si genera il segnale Qα

p_2 si dispone il FF ΔN ed il FF D

p_3, p_4, p_5, p_6 i caratteri inviati dall'U.C. vengono staticizzati nelle rispettive decadi del contatore di blocchi del GUN

p_7 il carattere inviato dall'U.C. viene staticizzato, solo per i bit a, b, c, nella decade delle decine di migliaia del contatore di blocchi del GUN

p_8	se il carattere inviato dall'U.C. in p_7 contiene il bit d si dispone il FF J
p.d.c. βw, p_0	si dispone il FF NL_k relativo al nastro n; si interroga il FF No; se l'unità a nastro n risulta occupata l'istruzione non può essere eseguita; se l'istruzione può essere eseguita si ripone il FF Nf e si genera il segnale $Q\beta$ che dà inizio all'esecuzione.

21.4. **Istruzione AV** (Xn XXXX T —)

Decodificato ΔSXM

Con tale istruzione si riavvolge indietro la bobina montata sulla unità a nastro n fino al termine del nastro.

L'istruzione AV occupa l'U.C. e il GUN solo durante la fase α di preparazione e per un periodo di cifra (p_0) della fase βw.
Elenchiamo le principali operazioni:

Nella U.C.

p.d.c. α, p_0, p_1, p_2	tutto avviene come nella TN
p_3, p_4, p_5, p_6	ad ogni p.d.c. si legge in Memoria e si rigenera in Memoria il carattere letto
p_7	si trasferisce da Memoria al GUN il carattere che individua il nastro n
p_8	col p.d.c. p_8 termina la fase α
p.d.c. βw, p_0	si dispone al mastro M_7 il FF φ; si ripone il FF βw; l'U.C. può eseguire le successive istruzioni.

Nel G.U.N.

p.d.c. α, p_0	si dispone il FF Nf
p_2	si dispone il FF Rv
p_7	col carattere inviato dall'U.C. e cioè il carattere che individua il nastro n, si interroga il FF No. Se il FF No si dispone, l'istruzione non viene eseguita.
p_8	si dispone il FF NR_{vk} relativo al nastro n. L'unità a nastro relativa si avvia all'indietro.

PARTE TERZA

SIMULTANEITÀ OPERATIVA E VARIE

22. SIMULTANEITÀ OPERATIVA

22.1. Considerazioni generali

Una generica istruzione dell'Elea 9003 si svolge in due fasi distinte; la fase preparatoria, durante la quale il canale interno è impegnato in continuità; la fase esecutiva che impegna sempre per un periodo di cifra (βw, p_0) il canale interno, prima che inizi l'esecuzione vera e propria e che potrà impegnare entrambi o uno solo dei canali, o anche nessuno dei due, impegnando però il GUN o il governo di una determinata unità « on line ».

È quindi possibile eseguire contemporaneamente più istruzioni che impegnino organi diversi.

Per ottenere una buona utilizzazione dei vari organi è stato reso possibile svolgere tre programmi contemporaneamente. Si deve notare che esiste un preciso criterio di precedenza fra i tre programmi; precisamente il 3° programma ha la precedenza assoluta, quindi ha la precedenza il 2°, per ultimo il 1°.

Le istruzioni interessanti il governo unità « on line », che si suppone già conosciuto, devono obbligatoriamente essere poste sul 3° programma. Per altre istruzioni non esistono limitazioni di sorta; ovviamente però i salti al secondo programma S_2P ed S_2P^* non possono essere nè sul 2° programma, nè sul terzo; così pure i salti al terzo programma S_3P ed S_3P^* non possono essere sul terzo programma.

Il terzo programma cede la precedenza agli altri qualora manchi il segnale UD inviato dal G.O.L. all'U.C., oppure venga letta una istruzione che non può essere eseguita; in tale ultimo caso si dispone il FF N_3. Analogamente il secondo programma cede la precedenza al primo se viene letta una istruzione che non può essere eseguita; in tal caso si dispone il FF Ne.

22.2. FF Ni, Ne, N_3

La funzione di tali FF è già stata descritta nel capitolo 1°; ricordiamo che si devono disporre quando una istruzione, letta rispettivamente sul 1°, 2° o 3° programma, non può essere eseguita; precisamente si hanno i seguenti casi:

a) βy (cioè canale esterno già occupato) e Δβy (per l'esecuzione dell'istruzione occorre il canale esterno); si dispone il rispettivo FF in α, p_5, M_4

b) $\overline{Nf}$ (cioè G.U.N. occupato) e ΔN (cioè istruzione di nastro); si dispone il relativo FF in α, p_5, M_4

c) $\overline{Nf}$ (cioè G.U.N. occupato) e ΔSXM (cioè istruzione AV, non compresa in ΔN); si dispone il relativo FF in βw, p_0, M_4

d) $\overline{Nf}$ (cioè G.U.N. occupato da una istruzione di nastro) e XEW (cioè salto condizionato a segnalazioni da G.U.N.); si dispone il relativo FF in βw, p_0, M_4

e) No* (cioè si è disposto il FF No del GUN) e ΔNS = ΔN + ΔSXM (è una istruzione di nastro, compresa l'AV); si dispone il relativo FF in βw, p_0, M_4

f) XFλ = ΔAUR + ΔAEB. (TAST) (istruzione che impegna la telescrivente) e Ts (cioè precedente istruzione MS non ancora terminata); si dispone il rispettivo FF in α, p_5, M_4

g) XFλ e ISU_2.($\overline{ISOL}$) (è ancora in corso il ritorno carrello dovuto alla precedente istruzione impegnante la telescrivente); il segnale ($\overline{ISOL}$) è dovuto al fatto che se vi è (ISOL) anche durante un ritorno carrello si può disporre il FF Ts e quindi preparare la MS; se vi è invece il segnale ($\overline{ISOL}$) per disporre il FF Ts si attende il segnale Tp, che si avrà solo alla fine del ritorno carrello, occorre perciò rileggere l'istruzione dopo la fine del ritorno carrello.

   Si dispone il relativo FF in α, p_5, M_4.

Per i FF Ni ed Ne vi è inoltre la condizione:

ΔSOL (decodificato dell'istruzione S_3P) e 3p (cioè terzo programma già in corso). L'istruzione S_3P non può essere eseguita. Il rispettivo FF si dispone in βw, p_0, M_4.

Per il solo FF Ni vi è ancora la condizione:

$I_2.I_9$ (decodificati dell'istruzione S_2P) e 2p (cioè secondo programma già in corso). Il FF si dispone in βw, p_0, M_4.

22.3. **FF Sb, Sb_3**

La funzione di tali FF è già stata descritta nel capitolo 1; ricordiamo che si devono disporre quando cessa una condizione che può aver impedito l'esecuzione di una istruzione letta rispettivamente sul 2° o sul 3°; e precisamente si hanno i seguenti casi in cui si dispongono entrambi i FF Sb ed Sb_3:

a) Quando si libera il canale esterno, e cioè si ripone il FF βy, con $R\beta y . M_8$. (Corrispondente al caso a del paragrafo precedente).

b) Quando si libera il GUN, e cioè si dispone nel GUN il FF Øem, con Øem δ. (Corrispondente ai casi b, c, d del paragrafo precedente).

c) Quando si libera una unità a nastro precedentemente occupata chè in riavvolgimento ($\overline{\varnothing RV\delta}$). (Corrispondente al caso e).

d) Alla fine di una MS, quando si libera la telescrivente, con $\tau . P_3 . \overline{T_S} . M_8$. (Corrisponde al caso f).

e) Se vi è $\overline{ISOL}$, cioè se contemporaneamente all'istruzione MS è permesso eseguire altre istruzioni, quando termina lo stato instabile dell'univibratore ISU_2. (Corrisponde al caso g).

Il solo FF Sb è inoltre disposto quando si ripone il FF 3p; ciò corrisponde alla condizione ΔSOL, 3p con cui si può arrestare lo svolgimento del 2° programma, cioè disporre Ne.

Assolta la loro funzione, e cioè ripreso rispettivamente il 2° o il 3° programma, i FF Sb ed Sb_3 si ripongono rispettivamente in $\alpha_2 . P_3$ o $\alpha p . P_3$.

22.4. **Criteri generali per i tre programmi**

Vi sono alcuni casi in cui, dopo aver eseguito una istruzione di un programma, sia esso il 1° o il 2° o il 3°, non è possibile eseguire un'istruzione di un altro programma, ma occorre eseguirne ancora una dello stesso.

In tali casi esiste il segnale $\Delta\alpha_2$ formato dall'or dei seguenti and:

a) ΔZ.ICV. Infatti ad un salto verificato deve immediatamente seguire l'istruzione a cui ha rinviato il salto; altrimenti l'indirizzo che la individua, contenuto nel registro J0, verrebbe perduto.

b) ΔPI. Il decodificato ΔPI è caratteristico delle istruzioni PIN, PUM; a queste deve ovviamente seguire immediatamente la rispettiva NAN o una ΔMM.

c) δPRC.ΔV.ΔU.ΔM. Cioè ad una PRN che può essere eseguita essendo libero il GUN, fatto segnalato dal disporsi del FF Prc del GUN, deve immediatamente seguire la relativa TN.

d) ΔZ.λ. Tale segnale è presente all'inizio della seconda fase α della SIN per evitare che dopo la prima fase α si passi ad un altro programma.

Se per qualsiasi motivo il 3° programma non può proseguire, la precedenza viene ceduta al 2°, se il secondo stesso esiste; se anche il 2° non può proseguire si passa ad eseguire le istruzioni del 1°; se anche sul primo si legge una istruzione non eseguibile si continua a rileggerla indefinitivamente; ad ogni fase α segue un solo p.d.c. βw, p_0 e quindi di nuovo una fase α.

Se invece il 1° programma è già finito, cioè è stata già letta sullo stesso l'istruzione di stop, e sia il 3° che il 2° non possono proseguire, l'U.C. rimane ferma con assenza di qualsiasi fase; la linea di ritardo però continua a ciclare.

La linea di ritardo viene arrestata solo quando tutti i programmi sono finiti.

22.5. Il terzo programma

Il sincronizzatore (vedasi la relativa descrizione) invia all'U.C. due segnali indicanti, se presente uno qualsiasi dei due, che il terzo programma per quanto riguarda il GOL, può eseguirsi; essi sono:

a) 1c Indica che si è durante il primo ciclo del 3° programma; tale segnale è dato da un FF disposto dall'istruzione S_3P, è pertanto presente alla fine dell'istruzione S_3P e causa la partenza del 3° programma assicurandone la continuazione durante il primo ciclo.

b) UD Il comparire di tale segnale indica che, per quanto riguarda le unità « on line », il terzo programma può riprendere quando si è arrestato per mancanza di 1c e di UD; la permanenza di tale segnale indica che il terzo programma può continuare per quanto riguarda il GOL.

Il segnale Sb_3 indica che il terzo programma, per quanto riguarda gli organi di U.C. e di GUN, può riprendere, se si è arrestato a causa della presenza di N_3.

Il segnale $\overline{N_3}$ indica che il terzo programma, per quanto riguarda gli organi di U.C. e di GUN, può continuare.

Si hanno pertanto le seguenti condizioni per il disporsi del FF αp:

a) $XFS . 1c . \overline{N_3}$ dove XFS è il consenso al disporsi del FF α.

Cioè si dispone il FF αp, contemporaneamente al FF α, se si è durante il primo ciclo, ed il terzo programma, per quanto riguarda l'U.C. ed il G.U.N. può continuare ($\overline{N_3}$).

b) $XFS . 1c . Sb_3 . \overline{\Delta\alpha_2}$

Tale and è utile nel caso in cui durante il primo ciclo il terzo programma si è arrestato, cedendo la precedenza agli altri, per il disporsi di N_3. Quando compare il segnale Sb_3 il terzo programma può riprendere immediatamente purchè sul programma che è stato svolto nel frattempo non sia stata letta una istruzione a cui deve seguirne un'altra dello stesso programma; in tal caso infatti è presente $\Delta\alpha_2$, e il terzo programma non può riprendere.

c) $XFS . 1c . Sb_3 . PIII$

Tale and è utile nel caso in cui durante il primo ciclo il terzo programma si è arrestato, per il disporsi di N_3, e però non sono state svolte istruzioni di altri programmi essendo questi già terminati. L'ultima istruzione letta sul terzo programma può aver causato la presenza del segnale $\Delta\alpha_2$ per cui non è sufficiente l'and del caso b. Per esempio può leggersi sul 3° programma una PUM mentre il canale esterno è già occupato; disponendosi N_3 il 3° programma si arresta; se gli altri programmi sono già terminati non viene svolta alcuna istruzione; all'arrivo di Sb_3 si deve riprendere il 3° programma pur essendo presente il segnale $\alpha\Delta_2$.

d) $XFS . UD . \overline{N_3} . (\overline{\Delta\alpha_2} + PIII + I_{12} . I_{13})$

Se si tiene presente che il segnale UD indica al suo apparire che si deve riprendere il terzo programma, e con la sua permanenza

che lo si deve continuare si vede che il consenso PIII serve per quest'ultimo caso; il consenso $\overline{\Delta\alpha_2}$ è necessario per la ripresa del terzo programma. Si deve inoltre notare che il segnale PIII è utile anche nei casi di ripresa del terzo programma analoghi a quello descritto al punto c.

Il segnale $I_{12}.I_{13}$ è presente nella S_3P^*; tale istruzione fa comparire il segnale UD; essendo un salto verificato, è presente $\Delta\alpha_2$; essendo letta non sul terzo programma è assente PIII; ciononostante deve riprendere il terzo programma e ciò è ottenuto appunto col segnale $I_{12}.I_{13}$.

e) $XFS . UD . Sb_3 . \overline{\Delta\alpha_2}$

Tale caso è perfettamente analogo al punto b per i cicli di terzo programma successivi al primo.

f) $XFS . UD . Sb_3 . PIII$

Tale caso è perfettamente analogo al punto c per i cicli di terzo programma successivi al primo.

Il FF αp si ripone non insieme ad α, ma in $\beta w.P_0.M_9$; ovviamente dato che in $\beta w.P_0.M_9$ può di nuovo esserci il consenso per il disporsi di α, e quindi di αp, in tal caso il FF αp non viene neppure riposto.

Nelle istruzioni tipo ΔI (IT, + IT, — IT, CIT) il FF αp viene riposto in $\beta w.P_3.M_9$, cioè un periodo di cifra dopo il FFα.

22.6. **Il secondo programma**

La presenza in calcolatrice di un secondo programma in corso di svolgimento è segnalata dal FF 2p. Tale FF si dispone in $\alpha.P_2.M_3$ della prima istruzione letta sul secondo programma; si ripone quando viene letta sul secondo programma l'istruzione di STOP. Il FF α_2 si dispone contemporaneamente al FF α se esistono contemporaneamente i seguenti consensi:

a) $XEF = 2p + (\Delta + \Delta EE^*)$; dove è $\Delta + \Delta EE^* = I_2 . I_9 + I_{13} . I_{11}$

Cioè se è già in corso un secondo programma (2p) oppure lo si deve iniziare con l'istruzione S_2P ($I_2.I_9$) o con la S_2P^* ($I_{13}.I_{11}$).

b) XFD; è $\overline{XFD} = s\alpha p + \Delta\alpha_2 . \overline{PII} . \overline{PIII} . \overline{\Delta + \Delta EE^*}$

Cioè non si dispone α_2 se è presente il consenso al disporsi di αp, $s\alpha p$ (si dà così la precedenza al 3° programma); non lo si dispone inoltre se è stata letta una istruzione che non può cedere la precedenza ($\Delta\alpha_2$) sul primo programma ($\overline{PII}$, $\overline{PIII}$) purchè non sia proprio una delle istruzioni di salto al 2° programma: in tal caso è presente $\Delta + \Delta EE^*$ e pertanto si permette il disporsi di α_2.

c) $\overline{Ne}$ o Sb - Tali FF hanno funzione analoga a quella di N_3 ed Sb_3 per il terzo programma.

Si ha pertanto il consenso al disporsi del FF α_2:

$XFE = XFS . XEK . XFD . \overline{Ne} + XFS . XEK . XFD . Sb$

Il FF α_2 si ripone in modo analogo al FF αp.

22.7. Alcuni esempi

Supponiamo che siano in corso di svolgimento tutti e tre i programmi.

Sul terzo programma si legge in fase α (è presente anche αp) una istruzione LNa: tale istruzione occupa il canale esterno (si dispone βy) ed il G.U.N. (si ripone Nf); in $\beta w.P_0$ della LNa si dispone nuovamente il FF α; αp rimane disposto.

L'istruzione successiva sia una SEN: essendo il G.U.N. occupato ($\overline{Nf}$) non può essere eseguita e pertanto in αp, P_5 si dispone il FF N_3.

In $\beta w.P_0.M_9$, mancando $\overline{N_3}$ e non essendoci ancora Sb_3, è assente il segnale $s\alpha p$; quindi si ripone il FF αp; essendo presente $\overline{Ne}$ si dispone il FF α_2, contemporaneamente al FF α.

L'istruzione così letta sul secondo programma, se occupa solo il canale interno, è eseguibile; si prosegue così a leggere ed eseguire istruzioni del secondo programma; se, prima che sia finita la LNa in corso di esecuzione per conto del terzo programma, si legge sul secondo programma un'altra istruzione che occupa il canale esterno, per esempio una PUM, in $\alpha_2.P_5$ si dispone il FF Ne.

In $\beta w.P_0.M_9$, mancando $\overline{Ne}$, si ripone il FF α_2 e si dispone il solo FF α.

L'istruzione così letta sul primo programma, se occupa solo il canale interno, è eseguibile; si prosegue così a leggere ed eseguire istruzioni del primo programma.

Nel frattempo la LNa in corso di esecuzione per conto del terzo programma termina ad un M_{10} di occupare il canale esterno. Con il consenso R βy al riporsi del FF βy si dispongono all'M_8 i FF Sb ed Sb_3.

Pertanto alla fine dell'istruzione in corso di esecuzione sul canale interno, disponendosi nuovamente il FF α, si dispone anche il FF αp (essendoci Sb_3), purchè l'ultima istruzione letta ed eseguita sul 1° programma non dia il segnale $\Delta\alpha_2$; in tale ultimo caso, malgrado la presenza di Sb_3, viene ancora eseguita una istruzione del 1° programma.

Si rilegge pertanto l'istruzione SEN del terzo programma; la rilettura di tale istruzione è assicurata dalla presenza del segnale N_3; infatti in α, αp, Po si trasferisce in W l'indirizzo contenuto nel registro JH; tale indirizzo è quello del carattere di P_2 dell'istruzione SEN; il segnale N_3 impedisce che avvenga nel W la conta +10, e causa in P_1, M_0 una conta +1; in tal modo in P_1 si rilegge il carattere di funzione della SEN. In α, αp, P_3 i FF N_3 ed Sb_3 vengono riposti; essendo però il GUN ancora occupato per il periodo dedicato all'arresto dei nastri magnetici, si dispone nuovamente il FF N_3 in α, αp, P_5.

In βw, P_0, si ripone il FF αp e si dispone, contemporaneamente al FF α, il FF α_2, per la presenza di Sb. Si rilegge pertanto l'istruzione PUM del secondo programma; la rilettura di tale istruzione è assicurata dalla presenza del segnale Ne; in α, α_2, P_3 i FF Ne ed Sb vengono riposti; l'istruzione PUM, essendo libero il canale esterno (βy riposto) può essere eseguita; si prosegue pertanto nell'esecuzione del secondo programma.

Non appena il GUN termina completamente la LNa invia in U.C. il segnale ∅emδ, corrispondente come temporizzazione ad un mastro M_2, e si dispongono nuovamente entrambi i FF Sb ed Sb_3. Si rilegge pertanto nuovamente l'istruzione SEN del terzo programma che potrà finalmente essere eseguita, essendo presente il segnale Nf dal GUN. Si continua così nell'esecuzione del terzo pro-

gramma. Il terzo programma può essere interrotto, oltre che per il disporsi del FF N_3, anche per l'assenza del segnale UD da sincronizzatore. In tal caso si procede in modo analogo a quanto già visto cedendo la precedenza al secondo programma; l'unica differenza è che, ricomparendo il segnale UD e riprendendo quindi il terzo programma, se ne continua l'esecuzione senza rileggere l'ultima istruzione.

Si prosegue nello svolgimento dei tre programmi passando dall'uno all'altro nel modo descritto.

Occorre analizzare i vari casi possibili in cui possono terminare i tre programmi.

A) **Il terzo programma termina per primo**

Con la lettura sul terzo programma dell'istruzione STOP sull'ultima delle unità « on line » attivate, non può più comparire il segnale UD e si ripone il FF 3p; il terzo programma è terminato. Si deve notare che riponendosi il FF 3p si dispone il FF Sb; ciò perchè il secondo programma può essersi bloccato, disponendo il FF Ne, per aver letto su di esso una istruzione S_3P mentre era già in corso un terzo programma; si deve pertanto disporre il FF Sb affinchè il secondo programma possa riprendere e rileggere l'istruzione S_3P, che essendosi riposto 3p, sarà ora eseguibile.

Comunque, ammettendo che non venga iniziato un altro terzo programma, si alternerà lo svolgimento del secondo e del primo nel modo già descritto; si hanno quindi due sottocasi:

A1) Il secondo programma termina prima del primo.

Leggendo l'istruzione STOP sul secondo programma con $\Delta ST.\beta w.\alpha_2.M_4$ si ripone il FF 2p; ΔST è il decodificato dell'istruzione STOP: $\Delta ST = I_4.I_5$. Mancando il segnale 2p non si può più disporre il FF α_2 per l'assenza del consenso XEK; si prosegue pertanto nell'esecuzione del 1° programma. Se sul 1° programma viene letta un'istruzione non eseguibile si dispone il FF Ni; l'effetto di tale FF è di far rileggere l'istruzione stessa finchè non sarà possibile eseguirla.

Il primo programma avrà termine con la lettura dell'istruzione STOP: con $\Delta ST.\beta w.P_0.\overline{\alpha_2}.\overline{\alpha p}.M_2$ si dispone il FF St. Al successivo M_9 non si può pertanto più disporre il FF α; il calcolatore ha terminato il lavoro. Disponendosi il FF St, se non è in corso di esecuzione una MS ($\overline{TS}$) nè una KN ($\overline{\delta KN}$), al mastro M_4 si ripone il FF Ie, e viene pertanto impedito il riciclo della linea di ritardo di U.C.

A2) Il primo programma termina prima del secondo programma.

Leggendo l'istruzione STOP sul primo programma si dispone il FF St. Si deve notare che per poter aver letto tale istruzione, il secondo programma deve aver ceduto la precedenza, e pertanto il FF Ne è disposto.

Alla fine dell'istruzione STOP non esiste il consenso XFS per il disporsi del FF α; la linea di ritardo di U.C. continua a ciclare; il canale interno è inattivo con assenza di fasi; è perciò presente il segnale $\overline{XFM}$.

Col disporsi del FF Sb il secondo programma può essere ripreso; si ha infatti il consenso al disporsi del FF α, e quindi anche del FF α_2. È $XFS = 2p.St.(CONT).\overline{XFM}.\overline{XFN}$.

Il secondo programma può arrestarsi e riprendersi anche più volte, sempre per il gioco dei FF Ne ed Sb, finchè non viene letta l'istruzione STOP posta sul 2° programma; con tale istruzione si ripone il FF 2p e si apre la linea di ritardo di U. C. come nel caso A1.

B) **Il secondo programma termina per primo**

Con la lettura sul secondo programma dell'istruzione STOP si ripone il FF 2p; non si può più disporre il FF α_2 e pertanto il secondo programma è terminato. Si alterna quindi lo svolgimento del terzo e del primo nel modo già descritto; anche qui si hanno i due sottocasi:

B1) Il terzo programma termina prima del primo.

Con la lettura sul terzo programma dell'istruzione STOP sull'ultima delle unità « on line » attivate, non può più compa-

rire il segnale UD e si ripone il FF 3p. Si prosegue quindi nell'esecuzione del 1° programma che terminerà come nel caso A1.

B2) Il primo programma termina prima del terzo.

Questo caso è analogo al caso A2; infatti è chiaro che per leggere l'istruzione STOP sul primo programma il terzo deve aver ceduto la precedenza perchè si è disposto il FF N_3 o perchè non si ha più il segnale UD. Pertanto, dopo la lettura dell'istruzione STOP sul primo programma, il canale interno rimane inattivo con assenza di fasi.

Il consenso al disporsi del FF α, per poter riprendere il terzo programma, è dato da:

$$XFS = 3p . \overline{2p} . St . (CONT) . \overline{XFM} . \overline{XFZ} . XNS . UD_1$$

dove è $XNS = Sb_3 + \overline{N_3}$; cioè XNS è presente se si è disposto il FF Sb_3 (è cessata la condizione che poteva aver arrestato il terzo programma) oppure se è presente $\overline{N_3}$, cioè il terzo programma si è bloccato per mancanza del segnale UD; è $UD_1 = UD + 1c$; infatti il terzo può riprendere se è presente anche UD o 1c.

Il terzo programma può arrestarsi e riprendersi anche più volte, per il gioco dei FF N_3 ed Sb_3 o per l'alternarsi della presenza ed assenza del segnale UD da sincronizzatore. Con la lettura dell'istruzione STOP sull'ultima delle unità « on line » attivate, si ripone il FF 3p e si apre la linea di ritardo di U.C.

C) **Il primo programma termina per primo**

Con la lettura dell'istruzione STOP sul primo programma si dispone il FF St. Si deve notare che per poter avere letto tale istruzione sia il terzo che il secondo programma devono aver ceduto la precedenza. Affinchè il terzo programma abbia ceduto la precedenza è necessario che si sia disposto il FF N_3 o che sia cessato il segnale UD; per il secondo programma deve essersi disposto il FF Ne.

Pertanto non si ha il consenso per il disporsi del FF α ed il canale interno rimane inattivo.

Appena cessa la condizione che aveva bloccato il secondo programma, disponendosi Sb si ha il consenso per il disporsi del FF α; se è presente il segnale UD si dispone anche il FF αp e si prosegue il terzo programma, altrimenti viene ripreso il secondo; si alterna così lo svolgimento dei due programmi rimanenti con eventuali periodi di inattività del canale interno. Quando uno dei due è terminato si ricade in uno dei casi già visti precedentemente.

23. LA MEMORIA t

23.1. Criteri di funzionamento

La Memoria t è una Memoria ausiliaria, costituita da tre celle, singolarmente indirizzabili, di 8 bits ciascuna.

I primi cinque nuclei di ogni cella (a, b, c, d, e) vengono utilizzati per memorizzare nell'ordine il contenuto dei seguenti FF: C $\neq$, C $<$, Ot, Om, IZ. I rimanenti tre nuclei rimangono inutilizzati. Le tre celle si riferiscono ciascuna ad uno dei tre programmi. Esse vengono usate per conservare e ripristinare, quando si passa da un programma all'altro per il gioco delle precedenze, il contenuto dei suddetti FF relativo ai vari programmi.

In α, P_0 di ogni istruzione si seleziona la cella relativa al programma su cui era posta l'istruzione precedente e si scrive in essa il contenuto dei cinque flip-flop.

Nell'ultimo p.d.c. della fase α (P_2 per le ΔI, P_8 per le altre) si seleziona la cella relativa al programma su cui è posta l'istruzione che si sta leggendo ed il suo contenuto viene scritto nei rispettivi FF, ripristinando così il contenuto che essi avevano al termine dell'ultima istruzione svolta per conto del programma che è stato ripreso.

Si deve notare che tale palleggiamento di dati tra FF e Memoria t avviene anche se si continua a svolgere sempre lo stesso programma.

La Memoria t costruttivamente è unita all'accumulatore ed ai registri T; pertanto l'ingresso e le uscite sono comuni. La selezione di tali celle avviene tramite i rispettivi segnali Pt_1, Pt_2, Pt_3.

Non è possibile usare la Memoria t contemporaneamente all'accumulatore ed ai registri T.

23.2. Selezione memoria t

Si ottiene la selezione della prima cella, collegata al 1° programma, disponendo il FF Pt_1 in α, P_0, M_0, se l'istruzione precedente era posta sul primo programma ($\overline{PII}$, $\overline{PIII}$), e col segnale $XAU = \alpha r.P_8 + \alpha r.P_2.\Delta I$, se l'istruzione che si sta leggendo è posta sul primo programma ($\overline{\alpha_2}$, $\overline{\alpha_p}$) al mastro M_0.

Analogamente si seleziona la seconda cella, disponendo il FF Pt_2 in α, P_0, M_0 se l'istruzione precedente era posta sul secondo programma (PII) e con XAU, M_0 se l'istruzione che si sta leggendo è posta sul secondo programma (α_2).

In modo perfettamente analogo si dispone il FF Pt_3.

23.3. Ingresso ed uscita memoria t

In α, P_0 manca il consenso al mastro di strobe A-T, e quindi anche per la Memoria t.

Pertanto non si hanno uscite dalla Memoria t relativa al programma su cui era posta l'istruzione precedente; vi si scrive il contenuto dei 5 FF.

Con il segnale XAU viene pilotata la cella relativa al programma su cui si sta leggendo l'istruzione; il mastro di strobe A-T è presente e pertanto le uscite della Memoria t sono staticizzate nei FF Au. Con $XAU . M_4 = rIz$ o con rIz + (SG) si ripongono tutti i cinque FF. Sempre con il consenso $\overline{XAU}$ le eventuali uscite della Memoria t, e cioè i bits immagazzinati nei FF Au, dispongono i rispettivi FF.

I FF Au sono riposti con XAU al mastro M_6.

24. CONDIZIONAMENTI DA CONSOLE

24.1. Condizioni per l'arresto dell'elaborazione

L'elaborazione deve arrestarsi alla fine dell'istruzione in corso, e ciò è ottenuto evitando che si disponga nuovamente il FF α, nei seguenti casi:

a) Se è presente da console il segnale (ASE), arresto su un salto su errore, e l'istruzione in corso è un salto su errore (I_3), verificato o no.

b) Se è presente da console il segnale (AS), arresto su un salto, e l'istruzione in corso è un salto (ΔZ).

c) Se è presente da console il segnale (ASN), arresto su un salto non verificato, e l'istruzione in corso è un salto non verificato ($\Delta Z.\overline{ICV}$).

In tutti e tre i casi a, b, c, si ottiene il segnale IAC. La presenza del segnale IAC inibisce, a causa dell'assenza del segnale $\overline{XFZ}$, il disporsi del FF α.

24.2. Program-stop

Si ricorda che, se è presente il segnale (PS) da console, l'elaborazione deve arrestarsi alla fine dell'istruzione i cui caratteri letti in fase α, eventualmente modificati (cioè le uscite di U.A. BUa), risultano uguali ai corrispondenti caratteri dell'istruzione impostata sulla console.

Se nell'istruzione impostata sulla console alcuni caratteri sono il « ? », si considera senz'altro verificata l'uguaglianza.

Esiste pertanto un confrontatore, Ci, dei bits uscenti da U.A. BUa, con i bits di console, BCn; tale confrontatore dà un'uscita nel caso si abbia disuguaglianza.

In α, P_0, M_5 di ogni istruzione si ripone un FF, detto Si. Nei successivi periodi di cifra, se si ha uscita del confrontatore Ci, si dispone al mastro M_9 il FF Si, purchè siano presenti i consensi $\overline{XSA}$ ed XSB; XSA decodifica il carattere « ? » da console, quindi se da console si ha tale carattere il FF Si non si dispone; XSB è sempre presente in fase α, ed inoltre nella fase βw delle ΔI, tranne nei p.d.c.

14.

in cui si ha in uscita da Memoria un carattere ÷ o #; inoltre non si può disporre il FF Si in P_0 e se è presente il segnale Ap, cioè nel secondo p.d.c. di un p-pulso eventualmente raddoppiato per effetto del FF Ar.

Pertanto, alla fine della fase α, se in uscita di U.A. si è avuto anche un solo carattere diverso dal corrispondente carattere dell'istruzione impostata su console, il FF Si è disposto.

Se il FF Si è rimasto riposto, la presenza di $\overline{Si}$ e (PS) impedisce che col segnale φ si disponga nuovamente il FF α; l'elaborazione è arrestata.

24.3. Istruzione manuale

Qualsiasi istruzione può essere impostata sulla console; ciò è ottenuto scrivendone i rispettivi caratteri nei FF di console; si hanno infatti $6 \times 8 = 48$ FF di console.

Per eseguire tale istruzione, deve essere presente il segnale (MAN) da console.

In tal caso si ha per la durata della lettura dell'istruzione il segnale Iα (MAN), presente per tutta la fase α, e per la fase βw delle ΔI. Con tale consenso i FF Up (Ud) vengono disposti dai rispettivi bits provenienti da console. Il segnale (MAN) ovviamente impedisce che la Memoria venga pilotata in fase α, e nella fase βw delle ΔI.

INDICE